BHARATI VIDYAPEETH
(DEEMED TO BE UNIVERSITY), PUNE

Faculty of Engineering & Technology
B.Tech (Computer Engineering)
New Syllabus

BHARATI VIDYAPEETH
(DEEMED TO BE UNIVERSITY) Pune.

Faculty of Engineering & Technology

Programme : B. Tech. Computer Engineering(2021 Course)

Course Structure & Syllabus

(Choice based credit systems-2021)

B.Tech (Computer Engineering) Semester I to VIII

Manual1.

Sr. No	Parameter/ Dimensions	Department Responsible
1	Executive Summary	Electrical
2	Curriculum Concept	Civil
3	Curriculum Preamble	Civil
4	Curriculum salient features	BSH
5	Curriculum Details	
5.1	Courses-Theory/Practical's/Tutorials/Units/Co-mapping and Engagement , University exam and internal assessment	Computer
5.2	Credit Concepts- Equivalence	Mechanical
5.3	Vocational Courses - Objective, Hrs./Cr/Methodology, Assessment type, Record, Format for credit allotment/ Credit certificate/ Singing authorities.	ECE
5.4	Industry Taught Courses - Objective/Credit/Hrs. methodology, Approval format for expenditure, Request format for experts, Acceptance, Agreement time table, Display, Assessment - Theory/ Practical, Record, Bill format, Payment record.	ETC
5.5	NPTEL Courses - Objective- Methodology As sessment- Certificate- Credit certificate -Competent authority - Record.	Electrical
5.6	Projects (I & II) - Objective- Hrs./Credit, Description of stage I & II, Assessment evaluation, Format for TW evaluation and oral evaluation.	Mechanical
5.7	Social activity, assessment format, credit allotment, credit certificate	IT
5.8	Research paper	Chemical
5.9	Internship	Chemical

Executive summary

Students pursuing engineering studies need to be well equipped and state of art with the latest technological trends and industrial requirements. To produce the students with high caliber and technically sound, enrichment in the curriculum content and various quality initiatives are needed. This is possible only when the students undergo studies with an updated and evolving curriculum to match global scenario.

Curriculum Development History

- *In ambits of Deemed University- 2000*
- *Curriculum of SPPU Accepted*
- *First Revision in 2004*
- *Second Revision in 2007*
- *Third Revision in 2011*
- *Fourth Revision in 2014*
- *Fifth Revision in 2018 was expected*

The proposed curriculum is developed to inculcate the advanced engineering skills to cope up with upcoming industrial and societal needs. Students will be imparted with advanced contents from respective field and innovative delivery methods.

To inculcate the advanced engineering skills and knowledge, branch specific courses have been introduced from the Sem – I itself. There are total 38 theory courses, 4 vocational courses, 3 MOOCs, 2 projects, technical research paper writing, no. of application software courses, no. of practical based courses, 6 Industry taught courses along with 60 days exclusive internship have been incorporated in the curriculum with 230 credits and 6500 Marks.

There will be collaboration with the prominent industries to execute the vocational courses. These industries will deliver the content and execute the hands-on session to inculcate the required engineering skills of particular course. Also, one course per semester will be entirely delivered by the expert/s from the industry of respective field for which blended teaching learning will be adopted.

Students will apply the knowledge of respective courses and develop the prototype/ model as a part of project based learning.

To give the experience of technical writing and research article, students have to develop the two projects in pre final and final year respectively and shall submit the research article to reputed journal for publication. This will inculcate research aptitude among students and will enhance the research profile of institute also. Incorporation of various practical based courses in respective discipline, will give hands on experience to students to understand the engineering concept in better way. Nowadays all practices and process in the field are being computerized and automated. Hence, it was pertinent to increase software content in the curriculum. It was demand from the industry that every engineer should be conversant with Software/Programming/Data analysis and automation process. Hence, courses to such as C, C++, Python, Machine Learning, Artificial Intelligence are added in curriculum of all discipline. Students who wish to develop their career in the IT field, significant courses related to computational engineering and application software have been incorporated in the curriculum of each discipline.

National Education Policy is insisting the Online and Digital Education and Ensuring Equitable Use of Technology. To inculcate the self-learning approach amongst the students, proposed curriculum has introduced Massive Open Online Courses to all the students to provide an affordable and flexible way to learn new skills, advance the career and deliver quality educational experiences at scale.

2. Curriculum Content

- Curriculum derived from Latin word 'Currere', which means a race course or runway on which one runs to reach a goal.
- Curriculum is the instructional and educative programme by following students achieve their goals, ideals and aspirational life.
- Curriculum is a standards based sequence of planned experiences,

  which students practice and achieve proficiency in content and applied learning skills
- Its confidence building process
- Its total learning experience of the individuals
- Its interactive system of instructions and learning with specific goals , contents, strategies, measurements and resources.
- The desired outcome of curriculum is successful transfer / development of knowledge, skills, and attitude.
- Curriculum should lead to transformation of student to contributory member of the society

We tried to develop curriculum , which will meet these concepts.

Curriculum is the outline of concepts to be taught to students to help them meet the content standards. Curriculum is what is taught in a given course or subject. It refers to an interactive system of instruction and learning with specific goals, contents, strategies, measurement, and resources. It is a course of study that will enable the learner to acquire specific knowledge and skills. A curriculum consists of the "roadmap" or "guideline" of any given discipline. Both the philosophy of teaching of the instructors as well as of the educational institution serve as two of the principles upon which a curriculum is based.

In Engineering, a curriculum is the combination of instructional practices, learning experiences, and students' performance assessment that are designed to bring out and evaluate the target learning outcomes of a particular course. It is the goals, assessments, methods, and materials used to teach a particular skill or subject and includes thinking under "skill.". The curriculum needs to be planned *and designed in such a way so as to sequentially improve students' knowledge and skills.*

Placement is an important parameter and outcome of a good curriculum, which satisfy the need of good placement. The written curriculum is a plan of what is to be taught so that the student gets good placement. For this , a variety of technical and non-technical courses that are required to complete a specific degree so as to help the student for placement are included in the curriculum. In addition to technical knowledge , it should also include social behaviors as well as content and thinking skills.

Overall, the curriculum should be such that it should develop a student in a good job seeker, good entrepreneur and also a good human being.

All the above aspects are taken care in the curriculum of B. Tech-2021 course. This will develop different abilities in a student.

3. Curriculum Preamble

The curriculum 2021 is formed such that it will develop different abilities in a student. It a combination of blended teaching learning process in which both online and offline teaching is a part of the curriculum. In order to develop affection towards the discipline a student has selected, core discipline courses are included right from first year. This will also help to give the overall idea about the branch / discipline to the student.

Interaction with the industry is increased in this curriculum by introducing two new concepts –

1. Vocational Course and 2. Industry Taught Course.

Vocational Course (VC), a student will able to develop a specific skill set from the relevant people/ agency from the industry. This will add in gaining new skill sets required by the industry. Such Vocational Courses are included from Semester III to Semester VI of the curriculum. Department also design vocational course relevant for the discipline, which add practical knowledge to students. The vocational courses should be discipline specific. 4 vocational courses and 8 credits are integrated with curriculum.

Industry Taught Courses (ITC) are the courses which will be taught by the people from industry who are experts in the relevant field, either partially or fully. This will provide a scope to students to gain the latest knowledge as used in industry and also to have direct one on one interaction with the industry. This will develop a confidence among the students. Such teaching by industry experts will be as per their availability, if required online and other than official college hours also. Thus, there is a blend of online and offline teaching, knowledge from academicians as well as from industry. Total six Industry Taught Courses are included in the curriculum.

Industry Internship of 60 days at the end of Semester VI integrated with curriculum, will also add to the interaction with the industry. A student will avail his training in industry or on site or in any design office or research organization as allotted to him/by the institute. A separate logbook will be maintained by the student during this period duly signed daily by the competent authority.

Project Based Learning is a part of almost each course of the curriculum. Small projects on relevant topics will be allotted to the students as a part of term-work *of that course. This will inculcate the habit of applying the knowledge learnt to solve practical problems.*

Two Projects are included in two stages, one in third year (Sem V and Sem VI) and the second in final year (Sem VII and Sem VIII). Improvement in Research, thinking ability and application of theoretical knowledge to develop practical ideas is the main purpose of these projects.

Publication of a research paper is the outcome expected from the Project work and as a motivation, separate credits are allotted for this. Students are expected to write research article based on Project-I in standard journals in final year. Guide for Project -I will help in writing the research article.

To develop the self studying, self-learning skills, each student has to join the MOOC/NPTEL courses and will get the certification of the respective course. This will also give him/her a chance to get the knowledge from teachers from well known institutes of national repute. Three such MOOC/NPTEL courses are included each in Semester III, Semester V and Semester VII and separate credits are allotted to it.

Various new courses are introduced in the curriculum thereby introducing the current and latest technology to students. Basic Science and Engineering Science course contents are designed to match the requirement of the specific disciplines.

Number of software related to that branch/ discipline are included as part of the curriculum. This will help the students to get good placement.

Few soft courses are introduced to non-circuit branches. This will give a soft feel to such branches and also to inculcate confidence among the students.

In addition to technical abilities, a student needs to be developed as a good human being. For this, he will complete social activities in Semester IV and Sem VIII.

Thus Curriculum-2021 satisfies the requirements of National Education Policy-2021.

"Knowledge, Skill, Behavior" are the three attributes that are inculcated in a student when he completes his B.Tech. course under Curriculum-2021.

Recommendations considered

- *UGC- Quality mandate*
- *National Education Policy (NEP)*

- *AICTE model curriculum*
- *Curriculum of International Universities*
- *Curriculum of Indian Universities*
- *Feedback from HR of industries called for placements*
- *Market perception*

Methodologies Adopted In Designing Curriculum (2021-22)

- *19 Basic Points for design of Curriculum*
- *Listing of common points (credits, marks, No. of courses, common courses, industry taught courses, vocational Programmes etc.)*
- *Conducted series of meetings*
- *Conducted in depth one on one discussions with HoDs*
- *Planned three workshops,*
- *Eminent experts from Industry, IITs, IISER, NIT, SPPU, Central Universities were invited for workshops*
- *First workshop - Course structure, Titles of courses, Industry taught courses, Vocational Courses.*
- *Second workshop - Content of first and second year courses*
- *Third workshop - Content of third and fourth year courses- (Planned)*

4. Salient features

- *Total 250 contact hours teaching are incorporated.*
- *Credit based 38 theory courses being offered to achieve global standards of quality.*
- *Curriculum offers practicals to more than 80 % (~ 30 theory courses) theory courses.*
- *Total 230 credits (6500 marks) are offered for the entire B. Tech. programme.*
- *Theory courses contains 60% of courses and 20% to practical courses.*
- *Tutorials (6 Credits), online courses (6 Credits), vocational courses (6 Credits), projects (18 Credits), internship (3 Credits), Research Publication (2 Credits) and social activities assigned (4 Credits) contains remaining 20% of credits*
- *Blended education policy is adopted considering its importance. 20% courses are taught in online mode.*
- *Incorporation of 6 industry taught courses is one of the important and strategic step.*
- *Adopting 4 vocational Programmes in cooperation with industries, renowned agencies, universities will improve skillsets of our students.*
- *60 days industrial internship to meet the requirements of industry.*
- *Including of 2 projects to enhance technical skills & self learning.*
- *Research paper based on Project-I will inculcate research aptitude among students.*
- *Project based learning practically for all courses will enhance the ability of application of knowledge and problem solving aptitude.*
- *NPTEL/ MOOC courses in online mode are introduced as integrated part of the course structure.*
- *To understand social responsibility and social activities of weightage of 4 credits are integrated part of the course structure.*
- *Quantitative Techniques and communication courses are introduced to enhance the analytical ability of students and address employability.*
- *Wide range of elective courses have been offered to provide the choice, to explore the knowledge in their domain of interest.*

Salient Features

Sr. No.	UGC (Quality mandate)/ NEP2020-Recommendations	Curriculum (2021-22)
1	**Learning Outcome-based Curriculum Framework (LOCF)**	a) Programme outcomes and course outcomes are being made ready
2	**Imparting Life Skills to Students.**	a) Quantitative techniques b) Communication skills c) Bridging gap with Industry by vocational courses d) Self learning by NPTEL/PBL/Two projects
3	**Social and Industry Connect**	a) 6 Industry taught courses b) 4 Vocational courses c) 60 days internship d) Time and credits for social activities
4	**Promotion of Research and the Creation of New Knowledge.**	a) Research publications based on projects b) Project based learning
5	**Blended Education**	a) 15% courses in online mode b) NPTEL/MOOC courses in online mode
6	**Technology Enabled Learning/Self Learning**	a) NPTEL/MOOCS
7	**Software Applications**	a) Programme specific softwares and Software application Courses

5. Curriculum Details

5.1. Courses-Theory/Practical's/Tutorials/Units/Co-mapping and Engagement

Courses-Theory/Practical's/Tutorials/Units/Co-mapping and Engagement, University exam and internal assessment

The B.Tech. 2021 offers Credit and Outcome based curriculum with total 230 credits, required for graduation with a Bachelors' degree (B.Tech). The Under-Graduate Programme (B.Tech) is of four years duration i.e of eight semesters (two semesters/year).

Engagement of Courses:

The courses in revised curriculum structure of B.Tech. program are categorized under Core courses, Elective courses, Engineering Science courses and Basic Science courses. These courses are taught to students by engaging them through lectures, practical or tutorials by respective course coordinators. From semester I to VI, there are five (lecture engaged and assessed) courses and in semester VII and VIII there are four (lecture engaged and assessed) courses which are mandatory. All the courses have varying hours of engagement and credits. Theory lecture engagement varies between 3 hours to 4 hours/week, practical engagement varies between 2 hours to 4 hours/week for the respective courses. The contents of every course is divided into six units. Each unit can be covered in 6 hours or 8 hours depending on the total allotted hours/week of lecture engagement for the respective course. Some courses are solely practical oriented. These courses will be only engaged through laboratory sessions.

Outcome Based Curriculum:

Planning and realization of teaching and learning related to outcome-based curricular model requires that initial element shall be an outcome. It serves as a basis for defining modes of evaluation and validation of outcomes. The curriculum defines the Course Outcomes (COs) and course objectives for every course. The outcomes are assessed through various activities and evaluation of learner's performance in various examination schemes i.e Theory/Practical/Oral/Term work.

Credit Calculation:

The course credits are computed based on the teaching hours per week for that course using the formula as mentioned below.

Credits earned by the Student = Credits earned in Theory (Th) + Credits earned in Practical (P) / Oral (O) + Credits earned in Tutorial (T)

Here, as mentioned above, the credit assignment for Th/P/O/T of any course is based on number of teaching hours of that course. It is as mentioned here:

Number of Credits for Theory (Th) courses = Number of classroom teaching hours per week for that course (1:1 correspondence)

Number of Credits for Practical (P) / Oral (O) courses = Number of laboratory hours per week for that course / 2 (0.5:1 correspondence)

Number of Credits for Tutorial (T) courses = Number of tutorial hours for that course (1:1 correspondence)

Example: If a course has 4 hours of classroom teaching, 2 hours of laboratory session and 1 hour of tutorial, then the credits assigned for that course will be 4(Th), 1(P/O) and 1(T) respectively.

Examination Pattern:

A) University Examination (UE)

The pattern for theory examination is of 60:40, where the learner can earn 60 Marks (maximum) through University Examination (UE) and 40 marks (maximum) are assigned for Internal Assessment (IA). For the UE of Practical/Oral assessment, the total marks allotted are 50. The laboratory assessment is divided into three assessment heads viz. Term work (TW), Practical (P) and Oral (O). The students will be assessed through TW or P or O or combination of any of these for the courses that have practical assessment. 25 Marks are assigned to TW/P/O each, so when a learner is assessed for practical through TW and P heads, he/she will be assessed for 50 marks.

B) Internal Assessment (IA)

The Internal Assessment (IA) for the respective courses will be performed through Unit Tests (UT) and Assignments. Total two UTs of 20 marks each will be *conducted and the average marks of these two UTs will be considered. Similarly, course coordinators will design the class assignments in terms of exercises, case studies, real world problems or mini projects, which the learners have to submit from time-to-time, as mentioned by the deadline of each assignment. While designing the assignment, the course coordinators will provide the assessment criteria to the learners and maximum score (marks) for the assignment as well. If there are multiple assignments, then the average of score (from score attained in all assignments) will be calculated and considered as IA marks. This way, the learner will be assessed for 20 marks (maximum) for assignments.*

Hence, total marks for UT and assignments are 20 each and so, IA will be of 40 marks. The score for IA is calculated as:

IA Score attained by learner (Max 40) = Average Score attained in UTs (Max 20) + Score attained in Assignments (Max 20)

5.2. Credit Concept: Equivalence

In CBCS 2021 Course structure, the allotment of credits are as follows:

Theory class of 1 hour: 1 Credit

Practical class of 2 hours: 1 Credit

Tutorial class of 1 hour: 1 Credit

Project, Research Paper & Social Activity: 1 Credit

5.3. *Vocational course*

Vocational learning opportunities play a important role in skill development and employability of student. Vocational courses are ways of implementation of theoretical knowledge in the practice.The importance of vocational development can largely be summed up as the difference between theoretical knowledge vs. practical skills. The vocational courses are based on the teaching of practical skills. These courses are designed to introduce the manual skills in the professional education in addition to the theory. These courses will serve as bridge courses for professional growth and career improvement.

Aims & objectives of vocational courses:

- *To provide students with technical knowledge and skills necessary for progressive education in engineering profession.*
- *To give a better understanding of the emerging of technology.*
- *To train the student with necessary skills leading to skilled personnel who will be enterprising and self-reliant.*
- *To enhance the skill of students for becoming self-sustained engineer.*
- *To reduce the mismatch between the demand and supply of skill man-power.*

In this curriculum at B.Tech Programme, there are four vocational courses introduced i.e. in Semester III, IV, V and VI. The courses offered at these semesters are as per the requirement of the programme.

Methodology:

The vocational courses shall be conducted in association with the companies through MoUs. The candidate shall be provided training in the industries in respective area. The training can also be given by the company experts in the college with appropriate infrastructure. Departments can design vocational programme/course as per employability skills for an engineer of respective discipline required. The student shall have to attend the training sessions for at least 4 hours per week. The training sessions shall be organized on weekends or on the extended hours of the college timing.

A faculty-in-charge will be appointed to monitor the functioning of the vocational *course as well as monitor the performance of the student for the said course.*

The student has to maintain proper record of the training attended throughout the semester and submit the report on the work carried out. The record has to be checked and signed by the faculty –in-charge.

Assessment:

The assessment of the performance of the candidate for the vocational courses shall be in the form of term work and oral. The term work and oral carry 50 marks. The candidate performance shall be evaluated based on the training undertaken by the candidate throughout the semester. Student shall give presentation of skills he learned through vocational courses followed by viva. External examiner for the same shall necessarily from relevant industry.

A total of 2 credits shall be allotted per vocational course per semester.

Certificate:

Every candidate shall be awarded a certificate after successful completion of the vocational course as per the rules & regulations.

The certificate shall be jointly signed by concerned authorities of college and the company.

5.4. Industry Taught Courses

PREAMBLE:

The concept of Industry Offered Courses enables bridging of technological gaps between students and state-of-the-art technologies used current in the industry.

OBJECTIVES: To

i. *Impart the state-of-art technology course existing in the industry.*

ii. *Expose students to application of technologies adopted by industry.*

iii. *Train students for solving real-world projects in respective industries by applying technical knowledge gleaned from an industry expert*

iv. *Make students draw benefit from the experience of veterans from industry. Knowledge sharing by industry experts.*

v. *Align student's mind-set towards industrial environment through the instructor from industry. Provide industry instructor lead courses.*

CREDIT/HRS.:

Percentage of Industry Taught Courses in the programme =................ %

METHODOLOGY:

A) *A faculty shall be appointed as course co-ordinator. Roles and responsibilities of Course coordinator are as follows:*

(i) *Act as a liaison between identified Industry expert and department.*

(ii) *Arrange schedule of lectures in consultation with identified Industry expert.*

(iii) *Keep record of students' attendance.*

(iv) *Collect feedback from students and suggest changes and modifications in lecture delivery method by industry subject expert.*

(v) *Keep record of Unit Test Performance and Practicals along with experts.*

(vi) *Organise visit to the industry relevant to the course.*

B) Execution:

(i) The Identified industry expert can conduct theory classes on weekends or as per convenience of Industry experts either through offline or online mode. The courses which are to be taught by expert from industry are already identified and confirmed in workshop-I

(ii) Practical sessions will be conducted by course coordinator. Panel of experts from Industry shall be identified to teach the course before the commencement of the respective semester and submitted for the approval of the Head of the Institution with financial layout.

BHARATI VIDYAPEETH

(DEEMED TO BE UNIVERSITY)

COLLEGE OF ENGINEERING, PUNE – 411043.

Approval format for Expenditure for Industry Taught Course

Date:

Name of the Department: ____

1. *Budgetary allocation for industry expert (As per Budget 2021-22) Please mention total amount (in Rupees) and other bifurcations, if made---------------- (to be filled at college level)*

2. *No. of Lectures (Industry offered Course wise / Subject wise) required with specific subjects:*

Sr.No.	Title of the course	Name of Department	Semester	Work Load per week	Details of Industry Expert(s)				Total Remuneration
					Name & Designation of Expert	Name of the company	Contact Details	Honorarium per lecture	
1									
2									
3									

Recommendation for Course Coordinator
Recommendation for HoD
Recommendation for Principal

3. *Total financial Outlay for honorarium of Faculty: (Industry taught courses-Subject wise): with number of lectures (in Hours) in UG sections*

Sr. No.	Name of industry Expert	Honorarium	Financial Outlay (in rupees)
1			
Total			

Signature of HoD *Signature of Principal*

Request format-To Industry Expert

To

……

Subject: Industry Taught Course (ITC) for B.Tech (…..) , Sem-_____

Dear Sir,

Greetings from Dept. of _________, Bharati Vidyapeeth (Deemed to be University) College of Engineering, Pune

Bharati Vidyapeeth (Deemed to be University) College of Engineering, Pune, BV(DU)COEP an AICTE approved institution, was established in the year 1983 and is a constituent unit of Bharati Vidyapeeth (Deemed to be University), accredited (3rd Cycle) with 'A+' grade by NAAC and NBA.

In the national arena, BV(DU)COE Pune has been among top 100 Engineering Colleges of India, consecutively for five years (99th ranking in 2020) by MHRD in June 2020. It has also been ranked 20th at national level by AICTE Internshala for internships. Our reputation as India's premier engineering institution is further enhanced by being honored with the Platinum category by AICTE-CII survey. College is proud to be ranked 11th across India by the prestigious magazine India Today. DATAQUEST a leading journal, ranked BV(DU)COEP in 3rd position amongst the Top 50 Private T - Institutes of India. The college ranked 17th position in the survey conducted by Times of India in 2019.

----Brief about dept-----

The course curriculum has a multi-dimensional approach, it not only implements a dynamic, qualitative, and evolved structure and syllabus, but also incorporates a good and healthy mix of theoretical and practical exposure. In this regards the institute promotes and encourages courses in line with industry expectations and forthcoming challenges which should ease the students for undergoing industry offered courses for practical exposure of applications of Education system. This is much required to bridge the gap between Industry and Academia and by promoting industry orientation for creating a complete industry ready professional.

To fulfil these objectives, curriculum design, which will be implemented from the academic year 2021-22, B.Tech. program includes 6 courses taught by industry experts. With reference to the subject mentioned above, we request you to teach… ….. Total…… number of lectures (60 min each) are required to be delivered. A blended learning, to be offered for the students through combining online or offline teaching wherever and whichever is best possible. Therefore, I request you to send acceptance letter, mode of teaching, convenient day and time slot to teach the said course. Enclosed please find herewith standard format for reply.

With Thanks and Regards,

Sign and stamp of Head, Dept of ________

Enclose:- Course content

Reply

To

The Principal

BV(DU)

COE,

Pune.

Subject: - Acceptance for delivering/ conducting lecture of the course -------- of B.Tech(-----), Sem(----).

Ref.: - Your letter ------------------ dated-

Dear Sir,

This has a reference of your letter mentioned above. It gives me immense pleasure to accept your invitation to deliver lectures in the said course. Following will be the time-table for the lecture.

Sr. No.	Title of Course	Time	Days						
			Mon	Tue	Wed	Thu	Fri	Sat	Sun

Sincerely

<Signature >

< Name of Expert>

BHARATI VIDYAPEETH
(DEEMED TO BE UNIVERSITY)
COLLEGE OF ENGINEERING, PUNE – 411043.

Date:

AGREEMENT TIME-TABLE

Name of department:

Name of industry taught course:

Sr. No.	**Day**	**Date**	**Time Slot**

(Name & sign. of HOD with date & stamp)

(Name & Sign. of Concerned Person)

BHARATI VIDYAPEETH
(DEEMED TO BE UNIVERSITY)
COLLEGE OF ENGINEERING, PUNE – 411043.

Industry Taught Courses (Assessment- Theory/ Practical)

- *One course coordinator should be appointed for the course. All documents related to assessment of the course will be maintained by the course coordinator.*
- *Total assessment of Industry Taught Course -Theory is of 100 Marks.*
- *Assessment of this course consists of Internal Assessment and End Semester Exam which carry 40 Marks and 60 Marks, respectively.*
- *Internal Assessment consists of assignments and mini projects.*
- *One real world project (mini project) is considered as part of Internal Assessment.*
- *Students should give presentation on given topic.*
- *Industry expert should set question papers.*
- *In case of practical exam, industry expert can take oral exam (may be online) and students will perform the experiments in the presence of course coordinator in the department.*

BHARATI VIDYAPEETH
(DEEMED TO BE UNIVERSITY)
COLLEGE OF ENGINEERING, PUNE – 411043.

B.Tech (Brach Name) Sem __

Title of ITC: - _________

Record of Lecture Taken

Sr. No.	Lecture No.	Unit no.	Date of Conduction	Topic Covered	No. of Students Attended	Sign

BHARATI VIDYAPEETH
(DEEMED TO BE UNIVERSITY)
COLLEGE OF ENGINEERING, PUNE – 411043.

Bill format for remuneration for Industry Taught Courses

(The bill should be submitted directly to the concerned department on or before 5th of every month)

1. *Name of industry expert:* ______________________________

   Company/Industry name:

2. *Name of the Department:* ______________________________
3. *Remuneration for the Month:* ______________________________
4.

Name of the Bank	Branch	A/C No.	IFSC

5. *Contact Details: -*

Email	Cell Phone No.

6. *Details of lectures delivered:*

Sr. No.	Title of the Course	Class	Date	No. of lectures	Total Remuneration (Rs./lecture)
			Total		

Date: ____________

____

Signature of the Industry expert

Certified that   has been appointed by the ----- dept as an industry expert for the course vide order No. ______________________

dated. ________________________ has delivered ____ lectures/taken classes during the month/ Sem ____________________

and is entitled to honorarium of Rs. __________________(@Rs. ------ /- per lecture/per day)

Course Coordinator: __________________________

____

Signature of the Head of the Department with Seal

Date:

Receipt: -

Received with thanks ₹-------- from BVDUCOE, Pune towards conduct of -------- lectures of the course ------------- of B.Tech(-------), Sem--- --.

Signature of Industry Expert

BHARATI VIDYAPEETH
(DEEMED TO BE UNIVERSITY)
COLLEGE OF ENGINEERING, PUNE – 411043.

Payment Record
(Copy to be maintained in the Department)

Sr. No	Name of Department	Name of course	Name of Industry Expert	Name of company	Email	Mo. No	Address	Amount	Remark/ check number transaction id

Encl:
1) College voucher copy
2) NEFT/RTGS copy

5.5 MOOCs Implementation

To inculcate the self-learning approach amongst the students, proposed curriculum has introduced Massive Open Online Courses to all the students. It will provide an affordable and flexible way to learn new skills, advance the career and deliver quality educational experiences at scale.

Also, National Education Policy is insisting the Online and Digital Education and Ensuring Equitable Use of Technology.

A massive open online course (MOOC) is an online course aimed at large-scale interactive participation and open access via the web. In addition to traditional course materials such as videos, readings, and problem sets, MOOCs provide interactive user forums that help build a community for the students, professors, and teaching assistants (TAs).

BV(DU)COE Pune is having active NPTEL local chapter-partnership. *Proposed curriculum has introduced three MOOCs at B.Tech Sem – III, Sem V and Sem VII with following objectives.*

1. *To provide e-learning through online web and video courses in Engineering by experts in the country in that subject.*
2. *To develop self-learning attitude in students.*
3. *To provide platform for knowledge enhancement of student's as per their area of interest.*
4. *To update students with advanced technologies.*
5. *To make the students more employable.*
6. *To prepare the students for competitive exams like GATE and also for higher studies.*

Methodology of Assessment:

- *Department shall publish list of NPTEL courses in every semester. Student can refer selected one of them in respective semester.*
- *Considering pre-requisite, proposed curriculum has provided with the various subject baskets as per the courses available.*
- *Students need to enroll for the course in each academic year as mentioned in the structure.*

- *Students need to attend all online lectures and complete all assignments as per schedule for registered course.*
- *Student will register and appear for exam conducted by NPTEL and shall submit the copy of course completion certificate received after passing the exam for registered course.*
- *Accordingly, the credits will be allotted to the student for respective MOOCs.*
- *Students have the flexibility to attempt the said course during the entire B.Tech Programme to earn the credits of respective MOOCs.*
- *NPTEL courses relevant to respective branch are only expected to select by students. Credits will not be awarded if general/ non engineering courses opted.*

5.6 Project I and II

Project Stage I Objectives:

Provide help to the students

- *In generating a new idea or modify existing system for solving societal, industrial and/or institutional problem.*
- *In review of literature that aligns with new idea and/or existing systems and clearly defining the problem*
- *In developing a workflow process/methodology for the desired system.*
- *In designing various components of the system assembly*
- *In developing a CAD model of the desired system.*
- *In writing the technical report based on the work completed*

Project Stage II Objectives:

Provide help to the students

- *In fabrication of the experimental setup/new system and/or purchase of standard components*
- *In pilot run and/or validation of new system for its performance*
- *In modifying the system if required to improve its performance.*
- *In detailed parametric studies of the modified system and analyzing the results*
- *In writing the technical report, research article and/or filing a patent.*

Particular	Hours per week	Credits allotted
Project I stage I	2	4
Project I stage II	2	4
Project II stage I	4	4
Project II stage II	4	6

Assessment & Evaluation:

For Project-I Stage I & II		
	Assessed through	**Marks**
Assessment Tools	Presentation 1	10
	Presentation 2	10
	Presentation 3	10
	Continuous Assessment by guide	10
	Final Project demonstration, presentation & viva voce (University Examination)	60
	Total Marks	**100**

For Project-II Stage I & II		
	Assessed through	**Marks**
Assessment Tools	Presentation 1	20
	Presentation 2	20
	Presentation 3	20
	Continuous Assessment by guide	20
	Final Project demonstration, presentation & viva voce (University Examination)	120
	Total marks	**200**

Minimum number of in-sem. project presentations: 03

Parameters for evaluation of project in University examination

1. *Idea of Project/Topic*
2. *Technical content*
3. *Innovation*
4. *Experimentation/Model development/Software development/Simulation development etc.*
5. *Participation as an Individual*
6. *Research Potential*
7. *Project Hardware/Software*
8. *Fabrication/Model/Equipment development*
9. *Data Analysis*
10. *Attendance*
11. *Timely completion*
12. *Report writing*
13. *Presentation*

Prepare a format for report card of indicating progress, assessment and progressive evaluation of the project. This progressive evaluation record (PER) is prerequisite for university examination.

Progressive Evaluation Record (PER) shall be submitted in the department at the end of the semester and made available at time of university examination.

Format for Internal Examination for Project- I & II

B.Tech (------), Sem--------

Roll No.	PRN	Name of student	Term Work Marks			
			Presentation-I (10%)	Presentation-II (10%)	Presentation-II (10%)	Continuous Assessment by Guide (10%)

Format for University Examination for Project- I & II

Roll No.	PRN	Name of student	Parameter for assessment of project and marks for examination													Total out of 100	Any five parameters out of remaining
			Idea of Project/ Topic	Technical content	Innovation	Experimentation/Model development/ Software development/ Simulation development etc	Participation as an Individual	Research Potential	Project Hardware/ Software	Fabrication/Model/Equipment development	Data Analysis	Attendance	Timely completion	Report writing	Presentation		
			10	10	10	10	10	10	10	10	10	10	10	10	10		

Out of 13 parameters, parameters no. 1,3,4,6 & 8 are mandatory and may be considered for assessment of the project. Each parameter will carry 10 marks for Project-I & 20 marks for Project-II.

5.7 Social Activities for the Learners

A) Introduction

The prime objective of Bharati Vidyapeeth (Deemed to be University) College of Engineering, Pune is holistic development of students. The learner achieves the status as "whole" when he/she has not only achieved success in academics but also has succeeded in bringing the nation up by connecting with socially left-out elements and bringing ray of hopes into their lives. In this respect, the new curriculum encourages the learner on the social activities. In this case, student's social activities are provided by the colleges, but not limited to them. Total of four credits assigned for these activities.

B) Objectives

a) To make people create balances, so they do not only focus on academic aspects, but there can also be other aspects to have in life.

b) To build better relationship with others.

c) To create great balance with the academic aspects.

d) To learn and understand society.

e) To develop the nature of help and enhance the ethical norms for behaviors.

f) Teamwork

C) Outcome of Social Activities:

The social activities make a good impact on learners. The learner:

a) Will be able to understand the needs of society.

It enables a learner to consider the perspective of other people and understand their needs by interacting with people from diverse backgrounds.

b) Will be able to understand different perspectives and engage other cultures.

Social events develop social skills and empathy—the outward-oriented dimensions of emotional intelligence (EQ). The interactions or conversations elicited by events helps students build relationships, understand different perspectives and engage other cultures. Social events provide an opportunity to expand one's social circle.

c) *Will be able to maintain positive outlook towards life.*

With high adaptability to diverse situations and a good level of understanding of other's opinions, socially aware learners are less likely to indulge in negative behavior. They are also less vulnerable to stressful situations and have fewer chances of getting involved in undisciplined behavior. These students also have a more positive outlook on life.

d) *Will be able to maintain good emotional health.*

Social activities keep the learners sharp and mentally engaged, and this is important to prevent the onset of serious diseases like dementia or Alzheimer. Connecting with others helps keep you in a positive mood, which in turn wards off depression by improving physical health and maintaining good emotional health as well.

D) Sample list of Social Activities (not limited to them)

a) *Organizing Educational Camps*

Educational camps may be organized for the socially and economically week elements, especially in rural areas or even in the slum areas of the city, by making them aware of the importance of education and their own human rights.

b) *Tree Plantation Drive*

There are so many health benefits to having plants around – like fresher air, improved emotional state, and reduction of illness in and around the society. Tree plantation in this respect plays a crucial role. Just planting the tree is not enough but it should be made to grow to its extent.

c) *Offer Helping Hand for Martyrs Family by Fundraisers*

Soldiers fight for our country, securing our borders. They don't think of their family and sacrifice their lives for us, and what we do for them? Packages are announced every time after the death of our worriers but rarely reaches them. Families keep waiting for years. In this regard, few of these forgotten families can be visited and a small helping hand can be lend to them, to make them lead their further life peacefully. Fundraising in this respect, is a great student society social idea. It is incredible to see how people can bring positive change if they work together. The youth can make a team with an *external organization to take part in a purposeful community event as mentioned above.*

d) *National Service Scheme*

It will help in the overall personality development of a learner by participating in projects that benefit the community. This extra-curricular activity is sponsored by the Ministry of Youth Affairs and Sports.

e) *Felicitations of People who have contributed to the society but now forgotten by the society*

There are so many intellectuals in our society who have achieved great heights in their field, who are stalwarts in different field but never came into limelight, their contribution is not recognized. Few of these can be invited publicly or visited at individual level by making a team and felicitate to appreciate their contribution towards the society or nation. Some of these stalwarts may be like Anand Kumar who teaches underprivileged students for IIT-JEE without a penny, Shekhar Naik who is the Captain of Indian Blind Cricket Team, Ranjeet Singh Desale who even being a rural teacher, is awarded by UNESCO with Global Teacher Prize, Ritu Biyani who fought cancer, traveled across the country to spread awareness.

f) *Street Play on Social Awareness*

This is also typically known as "Nukkad Natak". This form has been used to propagate social and political messages and to create awareness amongst the people regarding social issues. What is important is that the plays make the people think. The play is seen by many people of different age groups who then question and discuss the contents of the play. There have been several plays exposing the mechanism of black marketing and hoarding. Some talk of the use of political power for pressurizing people. Others highlight caste conflicts or ideas about hygiene and health. Street plays are also used to encourage literacy amongst villagers. Street plays on some of the topics like degradation of Indian media, hypocrisy, responsibility towards environmental concerns, brain drain, dilapidated educational structure, safety issues and rights for women. child labor, organ/human trafficking etc., can be thought of. The learners can participate in street play festivals like Manthan Mahotsav, the largest street play festival in India.

g) *Poster Exhibition on Contributions of Heroes of India*

The learners can organize an exhibition to not only display but explain the contribution of Indian Heroes who have been forgotten and remained in the book of history. Some of these inspiring heroes may be Mihir Sen, Khashaba Dadasaheb Jadhav, Anandibai Joshi – First woman doctor from India, Bhikaji Cama, Khudiram Bose, Baba and Prakash Amte etc. Such exhibitions make inspired, the youth of today's generation.

h) *Waste Clean Drive*

i) *Educating literacy-poor societies about disposal of nature-harming objects*

j) *Distributing needful items for living in economically backward societies*

k) *Organizing early completion on national issues.*

l) *Cleaning of Public Places/ Traffic Management/ Police Mitra.*

m) *Organizing activities under engagement of people with Science and Technology.*

Report of social activities conducted each student shall be prepared in standard format. Appropriate documentary evidences shall be part of report of students correspondence with respective authorities for social activities, permissions, certificates from Institutes/Organization/Local Government are essential documents for award of credits under this head.

E) *Summary*

Thus the interactions or conversations elicited by such social events help students to build relationships, understand different perspectives and engage other cultures and these events not only will uplift the moral of the society but also ignite minds of generations ahead to provide their support and enthusiastically participate in such activities. Such interactions will certainly provide an opportunity to expand their own social circle.

5.8 Internship

Internship of 60 days is incorporated as an integrated part of curriculum structure-2021. The primary objective of internship is to make students familiar with industry environment and to take up on- site assignment as trainees or interns in order to bridge the gap between theory and industrial practices. It is mandatory for students to undergo in-plant training after completion of semester VI in reputed industrial organization. The student shall submit the "Intern Certificate" issued by the industry organization as well as a technical report not exceeding 30 pages within the stipulated time to be eligible for making a presentation before the committee constituted by the department. On the basis of daily work carried out in the industry, student shall prepare a record book. This record book shall be checked and signed by his/her supervisor from the industry where he/she is doing internship on daily basis.

University examination carries 50 marks and after successful completion, student may be awarded 3 credits for the internship work. Standard format for record book shall be as below. Marks will be awarded out of maximum 50 and three credits will be given upon completion of internship towards the degree requirements, as per the regulations. Internship will ultimately assist students to apply theory learned in classroom to industrial practices so as to understand engineering/technical solutions in a global, economic, environmental and societal context.

5.9 Research paper publication

Research paper publication is one of the innovative features of programme curriculum- 2021.

1. It has been & introduced in 7th semester. Two credits are awarded for the same subject to publish of research paper. Student shall publish a research paper in peer reviewed/ Standard journal(not in paid journals) based on research work carried out for Project-I. Guide for Project-I shall be responsible for Writing manuscript, Selection of journal for publication, Submission of manuscript to the journal. Progress report of publication of research paper shall be prepared in standard format and submitted for the award of credits. Students shall be first author of research papers. No name either of faculty members except guide or other students shall be added without any contribution in research/project work. Format for progress report of research paper published (To be maintained by Guide). A departmental committee comprising of head of department, project guide, and one senior professor will review the progress of this activity periodically (not exceeding three months). The suggestions/comments offered by committee will be incorporated in due course of time to accomplish the task within a predetermined period.

2. Research paper publication as a integrated part of the course structure, will inculcate research aptitude among students. This will help there in seeking admissions in reputed International Universities for higher studies. Further, this research aptitude developed may enhance his employability also.

3. This activity is expected to generate 15 to 20 publication per year, which will enhance research profile of department and institute too.

4. Hence, there should be team of maximum 3 to 4 students per project except very exceptional projects. Prior permission to increase team size is essential.

Weekly progress report of the research paper publication.

Title of the project -

Name of the Guide -

Weekly schedule of meeting- Day------- Time-----

Student Details - Name--------- PRN------ Roll No.---

Sr. No.	Week No.	Date	Work completed/done by students per week

Bharati Vidyapeeth

(Deemed to be University)

Pune, India

College of Engineering, Pune

B.Tech. (Computer Engineering) Program Curriculum

(2021 Course)

VISION OF UNIVERSITY:

Social Transformation Through Dynamic Education.

MISSION OF UNIVERSITY:

- To make available quality education in different areas of knowledge to the students as per their choice and inclination
- To offer education to the students in a conducive ambience created by enriched infrastructure and academic facilities in its campuses.
- To bring education within the reach of rural, tribal and girl students by providing them substantive fee concessions and subsidized hostel and mess facilities
- To make available quality education to the students of rural, tribal, and other deprived sections of the population

VISION OF THE INSTITUTE:

To be World Class Institute for Social Transformation Through Dynamic Education.

MISSION OF THE INSTITUTE:

- To provide quality technical education with advanced equipment, qualified faculty members, infrastructure to meet needs of profession and society.
- To provide an environment conducive to innovation, creativity, research, and entrepreneurial leadership.
- To practice and promote professional ethics, transparency and accountability for social community, economic and environmental conditions.

VISION OF THE DEPARTMENT

To pursue and excel in the endeavour for creating globally recognised Computer Engineers through Quality education.

MISSION OF THE DEPARTMENT

- To impart engineering knowledge and skills confirming to a dynamic curriculum.
- To develop professional, entrepreneurial & research competencies encompassing continuous intellectual growth.
- To produce qualified graduates exhibiting societal and ethical responsibilities in working environment.

PROGRAM EDUCATIONAL OBJECTIVES

The students of B.TECH. (Computer Engineering), after graduating with Bachelor of Technology degree in Computer Engineering, will able to

1. Demonstrate technical and professional competencies by applying Engineering fundamentals, computing principles and technologies.
2. Learn, practice and grow as skilled professionals/entrepreneur/researchers adapting to the evolving computing landscape.
3. Demonstrate professional attitude, ethics, understanding of social context and interpersonal skills leading to a successful career.

PROGRAM SPECIFIC OUTCOMES

1. To design, develop and implement computer programs on hardware towards solving problems.
2. To employ expertise and ethical practise through continuing intellectual growth and adapting to the working environment.

PROGRAM OUTCOMES

a. Apply the knowledge of mathematics, science, engineering fundamentals, and computing for the solution of complex engineering problems.
b. Identify, formulate, research literature, and analyse complex engineering problems reaching substantiated conclusions using computer engineering foundations, principles, and technologies.
c. Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for public health and safety, and cultural, societal, and environmental considerations.
d. Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
e. Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.
f. Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues, and the consequent responsibilities relevant to the professional engineering practice.
g. Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and the need for sustainable development.
h. Apply ethical principles while committed to professional responsibilities and norms of the engineering practice.
i. Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings
j. Communicate effectively on complex engineering activities with the engineering community and with the society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
k. Apply the engineering and management principles to one's work, as a member and leader in a team.
l. Recognise the need for and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

CORELATION BETWEEN GRADUATE ATTRIBUTES AND PROGRAMME OUTCOMES

Graduate Attributes/ Programme Outcomes	a	b	c	d	e	f	g	h	i	j	k	l
Engineering Knowledge	✓											
Problem Analysis		✓										
Design/Development of Solutions			✓									
Conduct Investigations of Complex Problems				✓								
Modern Tool Usage					✓							
The Engineer and Society						✓						
Environment and Sustainability							✓					
Ethics								✓				
Individual and Teamwork									✓			
Communication										✓		
Project Management and Finance											✓	
Life-Long Learning												✓

DEFINITION OF CREDITS:

1 Hour Lecture (L) per week	1 credit
1 Hour Tutorial (T) per week	1 credit
2 Hour Practical (P) per week	1 credit
4 Hours Practical (P) per week	2 credits

STRUCTURE OF UNDERGRADUATE ENGINEERING PROGRAMME

Sr. No.	Category	Breakup of Credits
1	Basic Science Course (BSC)	32
2	Engineering Science Course (ESC)	10
3	Core Courses (CC)	136
4	Elective Courses (EC)	10
5	Project (PROJ)	09
6	Internship (INT)	03
7	Vocational Course (VC)	04
8	Massive Open Online Course (MOOC)	04 (Add On)
9	Research Paper Publication (Research)	02 (Add On)
10	Social Activities (SA)	04 (Add On)
11	Mandatory Course (MC)	Non-Credit
12	Internal Assessment (IA)	-
13	End Semester Examination (ESE)	-
TOTAL		**200 (10 Add on Credits)**

DISTRIBUTION OF COURSE COMPONENTS

Sr. No.	Category	Number of Courses
1	Basic Science Courses (BSC)	08
2	Engineering Science Course (ESC)	02
3	Core Courses (CC)	34
4	Elective Courses (EC)	02
5	Project (PROJ)	02
6	Internship (INT)	01
7	Vocational Courses (VC)	04
8	Massive Open Online Courses (MOOC)	02
9	Research Paper Publication (Research)	01
10	Social Activities (SA)	02
11	Mandatory Courses (MC)	01
TOTAL		**59**

Program: B.TECH. (Computer Engineering) Semester – I 2021 Course

Sr. No.	Course Code	Name of Course	Teaching Scheme (Hrs./Week)			Examination Schemes (Marks)						Credits			
			L	P	T	ESE	IA	TW	OR	PR	Total	L	P	T	Total
1		Mathematics for Computing - I	3	-	1	60	40	-	-	-	100	3	-	1	4
2		Organic and Electrochemistry	3	2	-	60	40	25	-	-	125	3	1	-	4
3		Digital Electronics	4	2	-	60	40	25	-	-	125	4	1	-	5
4		Classic Data Structures	4	2	-	60	40	25	-	25	150	4	1	-	5
5		Computational Thinking and Programming Concepts	4	2	-	60	40	50	-	50	200	4	1	-	5
6		Programming Technologies and Tools Laboratory– I	**1**	**2**	-	-	-	25	-	25	50	1	1	-	2
			19	**10**	**1**	**300**	**200**	**150**	**-**	**100**	**750**	**19**	**5**	**1**	**25**

Program: B.TECH. (Computer Engineering) Semester – II 2021 Course

Sr. No.	Course Code	Name of Course	Teaching Scheme (Hrs./Week)			Examination Schemes (Marks)						Credits			
			L	P	T	ESE	IA	TW	OR	PR	Total	L	P	T	Total
1		Mathematics for Computing - II	3	-	1	60	40	-	-	-	100	3	-	1	4
2		Physics for Computing Systems	3	2	-	60	40	25	-	-	125	3	1	-	4
3		Numerical Computation	4	-	-	60	40	-	-	-	100	4	-	-	4
4		Electrical Technology	4	2	-	60	40	25	-	-	125	4	1	-	5
5		Paradigms of Programming	4	2	-	60	40	25	-	25	150	4	1	-	5
6		Programming Technologies and Tools Laboratory - II	**1**	**2**	-	-	-	25	-	25	50	1	1	-	2
7		Computer System Workshop Technology	-	2	-	-	-	50	-	50	100	-	1	-	1
			19	**10**	**1**	**300**	**200**	**150**	**-**	**100**	**750**	**19**	**5**	**1**	**25**

Program: B.TECH. (Computer Engineering) Semester – III 2021 Course

Sr. No.	Course Code	Name of Course	Teaching Scheme (Hrs./Week)			Examination Schemes (Marks)						Credits			
			L	P	T	ESE	IA	TW	OR	PR	Total	L	P	T	Total
1		Discrete Mathematics and Applications	3	-	1	60	40	-	-	-	100	3	-	1	4
2		Data Structures and Algorithmic Thinking	4	2	-	60	40	25	-	25	150	4	1	-	5
3		Computer Organisation and Design	4	-	-	60	40	-	-	-	100	4	-	-	4
4		Computer Networks	3	2	-	60	40	25	-	25	150	3	1	-	4
5		Software Engineering*	4	2	-	60	40	25	25	-	150	4	1	-	5
6		Programming Technologies and Tools Laboratory – III	1	2	-	-	-	25	-	25	50	1	1	-	2
7		Vocational Course - I	-	2	-	-	-	25	25	-	50	-	1	-	1
			19	**10**	**1**	**300**	**200**	**125**	**50**	**75**	**750**	**19**	**5**	**1**	**25**
		Social Activity – I	-	-	-	-	-	-	-	-	-	-	-	-	2

*** Industry Taught Course – I**

Note: List of Vocational Courses will be published by the department before the commencement of respective semester.

Program: B.TECH. (Computer Engineering) Semester – IV 2021 Course

Sr. No.	Course Code	Name of Course	Teaching Scheme (Hrs./Week)			Examination Schemes (Marks)						Credits			
			L	P	T	ESE	IA	TW	OR	PR	Total	L	P	T	Total
1		Probability and Statistics	3	-	-	60	40	-	-	-	100	3	-	-	3
2		Models of Computation	3	-	1	60	40	-	-	-	100	3	-	1	4
3		Computer Operating System	4	2	-	60	40	25	-	25	150	4	1	-	5
4		Database Management System	4	2	-	60	40	25	-	25	150	4	1	-	5
5		Wireless Communication*	4	2	-	60	40	25	25	-	150	4	1	-	5
6		Programming Technologies and Tools Laboratory – IV	1	2	-	-	-	25	-	25	50	1	1	-	2
7		Vocational Course- II	-	2	-	-	-	25	25	-	50	-	1	-	1
			19	**10**	**1**	**300**	**200**	**125**	**50**	**75**	**750**	**19**	**5**	**1**	**25**
		MOOC - I #	-	-	-	-	-	-	-	-	-	-	-	-	2

*** Industry Taught Course – II**

 # Add-on Course - List of MOOC and Vocational Courses will be published by the department before the commencement of respective semester.

Program: B.TECH. (Computer Engineering) Semester – V 2021 Course

Sr. No.	Course Code	Name of Course	Teaching Scheme (Hrs./Week)			Examination Schemes (Marks)						Credits			
			L	P	T	ESE	IA	TW	OR	PR	Total	L	P	T	Total
1		Algorithm Design and Analysis	3	2	-	60	40	25	-	25	150	3	1	-	4
2		Computer and Information Security*	4	2	-	60	40	25	25	-	150	4	1	-	5
3		Compiler Engineering	4	-	-	60	40	-	-	-	100	4	-	-	4
4		Data Warehousing and Mining	3	2	-	60	40	25	-	25	150	3	1	-	4
5		Microprocessors and Microcontrollers	4	2	-	60	40	-	-	-	100	4	1	-	5
6		Programming Technologies and Tools Laboratory – V	1	2	-	-	-	25	-	25	50	1	1	-	2
7		Vocational Course- III	-	2	-	-	-	25	25	-	50	-	1	-	1
			19	**12**		**300**	**200**	**125**	**50**	**75**	**750**	**19**	**6**	**-**	**25**
		Social Activity – II	-	-	-	-	-	-	-	-	-	-	-	-	2
		Environmental Studies**	2	-	-	50	-	-	-	-	-	-	-	-	-

*** Industry Taught Course – III**
Note: List of Vocational Courses will be published by the department before the commencement of respective semester.

**** Mandatory Audit Course**

Program: B.TECH. (Computer Engineering) Semester – VI 2021 Course

Sr. No.	Course Code	Name of Course	Teaching Scheme (Hrs./Week)			Examination Schemes (Marks)						Credits			
			L	P	T	ESE	IA	TW	OR	PR	Total	L	P	T	Total
1		Big Data Analytics	4	2	-	60	40	25	-	25	150	4	1	-	5
2		Essentials of Internet of Things	4	2	-	60	40	25	25	-	150	4	1	-	5
3		Graphics Techniques and GPU	4	2	-	60	40	25	25	-	150	4	1	-	5
4		Quantitative Techniques, Communication and Values	4	-	-	60	40	-	-	-	100	4	-	-	4
5		Mobile Architecture and Programming*	4	2	-	60	40	25	-	25	150	4	1	-	5
6		Vocational Course- IV	-	2	-	-	-	25	25	-	50	-	1	-	1
			20	**10**		**300**	**200**	**125**	**75**	**50**	**750**	**20**	**5**	**-**	**25**
		MOOC - II #	-	-	-	-	-	-	-	-	-	-	-	-	2

*** Industry Taught Course – IV**

 # Add-on Course - List of MOCC and Vocational Courses will be published by the department before the commencement of respective semester.

Program: B.TECH. (Computer Engineering) Semester – VII 2021 Course

Sr. No.	Course Code	Name of Course	Teaching Scheme (Hrs./Week)			Examination Schemes (Marks)						Credits			
			L	P	T	ESE	IA	TW	OR	PR	Total	L	P	T	Total
1		Artificial Intelligence*	3	2	-	60	40	25	-	25	150	3	1	-	4
2		Virtualisation and Cloud Computing	3	2	-	60	40	25	25	-	150	3	1	-	4
3		Scalable Computing	3	2	-	60	40	25	25	-	150	3	1	-	4
4		Elective - I	4	-	1	60	40	-	-	-	100	4	-	1	5
5		Programming Technologies and Tools Laboratory – VI	1	2	-	-	-	-	-	50	50	1	1	-	2
6		Project Stage - I	-	2	-	-	-	50	50	-	100	-	3	-	3
7		Internship	-	-	-	-	-	25	25	-	50	-	3	-	3
			14	**10**	**1**	**240**	**160**	**150**	**125**	**75**	**750**	**14**	**10**	**1**	**25**

*** Industry Taught Course – V**

Elective – I	Software Testing and Quality Assurance	Mobile Operating System	Fundamentals of Fog and Edge Computing	System Thinking

Program: B.TECH. (Computer Engineering) Semester – VIII 2021 Course

Sr. No.	Course Code	Name of Course	Teaching Scheme (Hrs./Week)			Examination Schemes (Marks)						Credits			
			L	P	T	ESE	IA	TW	OR	PR	Total	L	P	T	Total
1		Machine Learning*	3	2	-	60	40	25		25	150	3	1	-	4
2		Data Storage Networking	4	-	-	60	40	-	-		100	4	-	-	4
3		Data Visualisation	3	2	-	60	40	25	-	25	150	3	1	-	4
4		Elective – II	4	-	1	60	40	-	-	-	100	4	-	1	5
5		Programming Technologies and Tools Laboratory – VII	1	2	-	-	-	25	-	25	50	1	1	-	2
6		Project Stage - II	-	4	-	-	-	100	100	-	200	-	6	-	6
			15	**10**	**1**	**240**	**160**	**175**	**100**	**75**	**750**	**15**	**9**	**1**	**25**
		Research Paper Publication#	-	-	-	-	-	-	-	-	-	-		-	2

*** Industry Taught Course – VI**
 # Add-on Course

Elective – II	Intelligent Autonomous Systems & Robotics	Deep Learning	Blockchain and Cryptocurrencies	Docker and Kubernetes

B.TECH. (Computer Engineering)

SEMESTER – I

COURSE SYLLABUS

Mathematics for Computing - I

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	03 Hours/Week	University Examination	60 Marks	Lecture	03
Tutorial:	01 Hours/Week	Internal Assessment	40 Marks	Tutorial	01
		Total	100 Marks	Total	04

Course Objectives:

- Linear equations and its basis and dimension.
- Linear mapping and its matrix representation.
- Orthogonalization and diagonalization of matrices.

Prerequisite:
Knowledge of algebra of matrices and determinants

Course Outcomes: On completion of the course, students will have the ability to:

1. Apply rank of matrix in solving system of equations.
2. Identify basis and dimension of matrix.
3. Solve problems on kernel and image of linear transformation.
4. Apply linear operator to represent matrix.
5. Evaluate orthogonalization of inner product space.
6. Use methods to find eigen values and eigen vectors.

Unit I System of Linear Equation **06 Hours**
Vectors and linear combinations, Rank of a matrix, Gaussian elimination, LU Decomposition, Solving Systems of Linear Equations using the tools of Matrices.

Unit II Vector Spaces **06 Hours**
Definition, linear combination, spanning sets subspaces, linear dependence and independence, basis and dimension, rank of matrix.

Unit III Linear Mapping **06 Hours**
Linear mapping, Kernel and image of linear mapping, rank and nullity of a linear mapping, singular and non-singular linear mapping.

Unit IV Linear mapping and matrices **06 Hours**
Matrix representation of linear operator, change of base, similarity matrices

Unit V Inner Product space and orthogonalization **06 Hours**
Inner product space, Cauchy-schwarz equality, Orthogonality, Orthogonal sets and bases, projections, Gramschidt orthogonalization, orthogonal and positive definite matrices, matrix representation of inner product

Unit VI Diagonalisation: Eigen values and eigen vectors **06 Hours**
Characteristic polynomial, Cayley-Hamilton theorem, eigen values and eigen vectors, properties.

Textbooks

1. P. N. Wartikar and J. N. Wartikar, Applied Mathematics (Volumes I and II), 7th Ed., Pune Vidyarthi GrihaPrakashan, Pune, 2013.
2. B. S. Grewal, Higher Engineering Mathematics, 42nd Ed., Khanna Publication, Delhi

3. B.V. Ramana, Higher Engineering Mathematics, 6th Ed., Tata McGraw-Hill, New Delhi, 2008.
4. Erwin Kreyszig, Advanced Engineering Mathematics, 10th Ed., John Wiley & Sons, Inc., 2015.

Reference Books

1. Peter V. O'Neil, Advanced Engineering Mathematics, 7th Ed., Cengage Learning, 2012.
2. Michael Greenberg, Advanced Engineering Mathematics, 2nd Ed., Pearson Education, 1998.

Project Based Learning - Provisional List of Projects

Students are expected prepare report on any one topic, write its definition, applications and illustrate with few examples. Also, write pseudo code for it, wherever applicable.

1. Gauss Elimination method.
2. LU-decomposition method
3. Rank of matrix
4. Linear combination
5. Basis and dimension
6. Spanning sets
7. Kernel and image of linear transformation
8. Rank-nullity theorem
9. Non-singular linear mapping
10. Linear operator
11. Similarity matrices
12. Change of base
13. Cauchy Schwarz equality
14. Orthogonality
15. Gram Schmidt Orthogonalization
16. Matrix representation of matrix
17. Cayley-Hamilton theorem
18. Eigen values and Eigen vectors

(Note: - *Students in a group of 3 to 4 shall complete any one project from the above list)

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Organic and Electrochemistry

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	03 Hours/Week	University Examination	60 Marks	Lecture	03
Practical:	02 Hours/Week	Internal Assessment	40 Marks		
		Term work	25 Marks	Practical	01
		Total	125 Marks	Total	04

Course Objectives:
The student should acquire the knowledge of

- To develop the interest among the students regarding chemistry and their applications in engineering.
- To develop confidence among students about chemistry, how the knowledge of chemistry is applied in technological field.
- The student should understand the concepts of chemistry to lay the groundwork for subsequent studies in the computing field.

Prerequisite:
Basic Chemistry

Course Outcomes: On completion of the course, students will have the ability to:

1. Differentiate between ionic and covalent bonding and classify the bonding in a compound as ionic or covalent.
2. Develop a working knowledge of the twelve fundamental principles of green chemistry and what it is all about.
3. Apply standard reduction potential data to determine the relative strength of oxidizing/reducing agents
4. Demonstrate the knowledge of polymer materials for futuristic engineering applications
5. Describe the properties of materials and Application of semiconductor electronics
6. Describe the manufacturing and refining process of fuels and lubricants

Unit I Chemical Bonding in Molecules **06 Hours**
MO theory, Structure, bonding and energy levels of bonding and shapes of many atom molecules, Coordination Chemistry, Electronic spectra and magnetic properties of complexes with relevance to bio-inorganic chemistry, organometallic chemistry.

Unit II Green Chemistry **06 Hours**
Introduction, Twelve Principles of Green chemistry, numerical on atom economy, synthesis, adipic acid and indigo. Organic dye- Traditional methods of organic dye. Green solvents (ionic liquid supercritical CO2), and products from natural materials.

Unit III Electrochemistry **06 Hours**
Electrochemical cells and Galvanic cells, EMF of a cell, Single electrode potential, Nernst equation, Electrochemical series, Types of electrodes, Reference electrodes, pH, pOH, acids and basis, Fuel cells, Construction and Working of - Acid and Alkaline Storage Battery, Dry Cell, Ni-Cd Batteries, Li-Ion Batteries, Li-Po Batteries.

Unit IV Polymers for The Electronics Industry **06 Hours**
Polymers, Conduction mechanism, Preparation of conductive polymers, Polyacetylene, Poly (p- phenlylene), Polyhetrocyclic systems, Polyaniline Poly (Phenylene sulphide), Poly (1,6-heptadiyne), Applications, Photonic applications.

Unit V Semi-Conductors, Insulators and Superconductors **06 Hours**
Semi conductivity in non-elemental materials, Preparations of semiconductors, Chalcogen photoconductors, photocopying process Introduction to Superconductors, types of Superconductors, Properties of superconductors, Applications of Superconductors, Electrical insulators, or Dielectrics.

Unit VI Fuels & Lubricants **06 Hours**
Classification of fuels, Calorific values, Comparison between solid, liquid, and gaseous fuels, Theoretical calculation of calorific value of a fuel, Selection of coal, analysis of coal, Natural Gas, Producer gas, water gas, Lubricants, Mechanism of lubrication, classification of lubricants, lubricating oils, Solid lubricants, Greases or Semi-Solid lubricants, Synthetic lubricants, Lubricating emulsions, Properties of lubricating oils.

Textbooks

1. Polymer Science and technology (2nd Edition), P. Ghosh, Tata McGRAW Hill, 2008.
2. Polymers: Chemistry & Physics of Modern Materials (2nd edition) J.M.G.Cowie, Blackie Academic & Professional, 1994.
3. A Textbook of Engineering Chemistry, Shashi Chawla, Dhanpat Rai & Co, 2004
4. Engineering Chemistry (16th Edition) Jain, Jain, Dhanpat Rai Publishing Company, 2013.

Reference Books

1. Inorganic Chemistry (4th edition), D. F. Shrives and P. W. Atkins, Oxford University, Oxford, 2006.
2. Reactions, Rearrangements and Reagents (4th edition), S. N. Sanyal, Bharti Bhawan (P & D), 2003.
3. Applications of Absorption Spectroscopy of Organic Compounds (4th edition), John R. Dyer, Prentice Hall of India Pvt. Ltd., 1978.

List of Assignments
Six assignments to be given by the course coordinator (Theory)-one from each unit/one mini project with report-students can work in group of 4 Maximum

List of Laboratory Exercises

1. Determination of Hardness of water sample by EDTA method.
2. Determination of Chloride content in water sample by precipitation titration method.
3. To determine strength of acid by pH – metric Titration
4. To measure the Conductance of a solution by conductometric titration
5. Measurement of Surface tension of a given liquid by Stalagmometer.
6. Determination of viscosity of a given liquid by Ostwald's Viscometer.
7. Determination of Saponification value of an oil sample.
8. To determine alkalinity water sample
9. Determination of Hardness of water sample by EDTA method.
10. Determination of Chloride content in water sample by precipitation titration method

11. To determine strength of acid by pH – metric Titration
12. To Prepare Phenol formaldehyde/Urea formaldehyde resin
13. To study set up of Daniel cell.

Project Based Learning - Provisional List of Projects

1. Green Chemistry approach to Nano-Structured Electronics
2. Assessment of Environmentally Benign Photopolymers as an Alternative to the Use of Formaldehyde Based Textile Finishing Agents
3. Solvent-Free Synthesis of Phthalocyanines
4. Synthesis of Conjugated Polymers and Molecules Using Sugar Reagents and Solventless Reactions
5. Environmentally Benign Control of Polymer Solubility: Photoresist Materials Using DNA Mimics
6. Enzymatic Synthesis of Non-Formaldehyde Phenolic Polymers: Control of Hydrogen Peroxide Concentration.
7. The materials chemistry and electrochemistry of lithium and sodium-ion batteries
8. Electroplating- the principles, how different metals can be used and the practical applications
9. Electroplating, Metal Polishing, Anodizing, Phosphating Metal Finishing and Powder Coating Projects.
10. To determine calorific value of a fuel by any suitable method
11. To study various properties of lubricants
12. To study various types of lubricants and its properties.
13. To determine quality of coal sample & its analysis.
14. To study mechanism of lubrication.
15. To study coal analysis & its significance.

Note: - Students in a group of 3 to 4 shall complete any one project from the above list)

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Digital Electronics

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	04 Hours/Week	University Examination	60 Marks	Lecture	04
Practical:	02 Hours/Week	Internal Assessment	40 Marks		
		Term work	25 Marks	Practical	01
		Total	125 Marks	Total	05

Course Objectives:

- To present the Digital fundamentals, Boolean algebra, and its applications in digital systems
- To familiarize with the design of various combinational digital circuits using logic gates
- To introduce the analysis and design procedures for synchronous and asynchronous sequential circuits
- To understand the various semiconductor memories and related technology

Prerequisite:
Mathematics and Elementary Physics

Course Outcomes: On completion of the course, students will have the ability to:

1. Comprehend different number systems and Boolean algebraic principles.
2. Apply logic design minimization techniques to simplify Boolean expressions
3. Analyse and design combinational logic circuits.
4. Demonstrate the operations of systems with sequential circuit elements.
5. Comprehend characteristics and structure of Programmable Logic Devices and Memory.
6. Draw ASM charts for sequential circuit design.

Unit I Digital systems **08 Hours**

Number Systems: Introduction to Number Systems-Decimal, Binary, Octal, Hexadecimal, Conversion of number system, Representation of Negative Numbers, 1's complement and 2's complement.
Binary Arithmetic: Binary addition, Binary subtraction, Subtraction using 1's complement and 2's complement, Binary multiplication, and division.
Digital Codes: BCD code, Excess-3 code, Gray code and ASCII code.
Logic Gates: Logical Operators, Logic Gates-Basic Gates, Universal Gates, realization of other gates using universal gates.

Unit II Logic Design Minimization **08 Hours**

Boolean algebra, De Morgan's Theorems, Standard representation of logic functions, Sum of Product (SOP) form, Product of Sum (POS) form, Simplification of logical functions, Minimization of SOP and POS forms using Karnaugh-Maps up to 4 variables Don't care condition, Quine-McCluskey Method.

Unit III Combinational Circuits **08 Hours**

Binary and BCD arithmetic, Half Adder, Full Adder, Half Subtractor, Full Subtractor, Binary Adder (IC 7483), BCD adder, Code converters
Multiplexers, De multiplexer, Decoder (IC 74138) and their use in combinational logic design, Priority Encoder, Digital Comparators, Parity generators and Checker (IC 74180), ALU.

Unit IV Sequential Circuits **08 Hours**

Flip- flop: SR, JK, D, T flip flops, Truth Tables and Excitation tables, Conversion from one type to another type of Flip Flop.
Registers: Buffer register, Shift register.
Counters: Asynchronous counters, Synchronous counters, Modulus counters

Unit V FSM and ASM charts **08 Hours**

Introduction to FSM, Moore and Mealy State machine, state machine as a sequential controller. Design of state machines: state table, state assignment, transition/excitation table, excitation maps and equations, logic realization, ASM chart notations, ASM block, State diagram, ASM chart for sequential circuits, Multiplexer Controller.

Unit VI Memory and PLD: **08 Hours**

Semiconductor memories: memory organization, memory expansion, Classification and characteristics of memories, RAM, ROM, EPROM, EEPROM, NVRAM, SRAM, DRAM.
Programmable logic devices: Study of PROM, PAL, PLAs. Architecture of PLA, designing combinational circuits using PLDs.

Textbooks

1. M. Morris Mano and M. D. Ciletti, Digital Design, Pearson Education.
2. RP Jain, Modern Digital Electronics, Tata McGraw Hill Publication.
3. F.J. Hill and G.L. Peterson, Switching Theory and Logic Design, John Wiley
4. J.F.Wakerly "Digital Design: Principles and Practices", 3rd edition, 4th reprint, Pearson Education, 2

Reference Books

1. David J. Comer, Digital Logic & State Machine Design, Oxford University Press.
2. Digital Integrated Electronics- H.Taub & D.Shilling, Mc Graw Hill.

List of Assignments

Six assignments to be given by the course coordinator (Theory)-one from each unit

List of Laboratory Exercises

1. Verify truth tables of logic gates. (AND, OR, XOR, NOT, NAND, NOR). Simplify the given Boolean expression using K-map and implement using gates
2. State De-Morgan's theorem and write Boolean laws. Implement NAND and NOR as Universal gates.
3. Design (truth table, K-map) and implement half and full adder/ subtractor.
4. Design (truth table, K-map) and implement 4-bit BCD to Excess-3 Code converters.
5. Study of magnitude Comparator using IC 7485
6. Implement of logic functions using multiplexer IC 74151 (Verification, cascading & logic function implementation)
7. Implement logic functions using 3:8 decoder IC 74138.
8. Verify truth tables of different types of flip flops.
9. Design (State diagram, state table & K map) and implement 3 bits Up and Down Asynchronous and Synchronous Counter using JK flip-flop
10. Design and implement modulo 'n' counter with IC 7490.

Project Based Learning - Provisional List of Projects

1. Survey report of basic gates ICs 7432, 4011, 4050, 4070,4071,40106
2. Implement combinational logic Circuit of given Boolean Equation.
3. Implement Half Adder and Half Subtractor.

4. Implement Full Adder using two Half Adders
5. Build 4-bit parallel Adder / Subtractor using IC.
6. Build Code Converters: Binary to Gray
7. Build Code Converters: Excess 3 to Binary)
8. Implement Two Bit Magnitude Comparator using IC 7485
9. Implement given combinational logic using MUX
10. Implement 7 segment decoder driver using IC 7447.
11. Build a Decade counter and Up-Down Counter.
12. Build a Shift Registers: SISO and SIPO
13. Implement the Johnson Counter and Ring Counter.
14.Survey Report on Static I/O and transfer Characteristic of TTL and CMOS.
15. Implement given Boolean Function using PLA. (Function and Equation will be given by Subject Teacher)

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Classic Data Structures

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	04 Hours/Week	University Examination	60 Marks	Lecture	04
Practical:	02 Hours/Week	Internal Assessment	40 Marks		
		Term Work	25 Marks	Practical	01
		Practical	25 Marks		
		Total	150 Marks	Total	05

Course Objectives:

The course focuses on enabling students to understand how data is stored in computer programs using data structures and facilitate them to use and build fundamental data structures.

Prerequisite:

Programming Basics

Course Outcomes: On completion of the course, students will have the ability to:

1. Compare and contrast the interfaces and internal representation of several linear abstract data types.
2. Solve given problems using array
3. Implement Stacks in a high-level programming language
4. Use and Implement Queues in a high-level programming language
5. Use and Implement lists in a high-level programming language
6. Demonstrate the ability to analyse, design, apply and use data structures and algorithms to solve engineering problems.

Unit I Introduction to Data structures & Arrays **08 Hours**

Need of Data structure, Classification of Data Structures, Static Data Organization, Operations on Data Structures, Abstract data Types (ADT).

Arrays: Introduction, Array Operations, representation of Arrays in Memory, Array with Functions, One- & Two-dimensional array in function, Implementation of One- & Two-Dimensional Arrays in Memory.

Applications: string handling, polynomial equation solving, sparse matrix multiplication, tic-tac-toe, and data visualization

Unit II Stacks **08 Hours**

Stack Definition and Structure, Operations on Stacks – create stack, Push stack, Pop stack, Stack top, Empty Stack, stack count, Destroy Stack, Array and Linked Representation, Types of Notations – Prefix, Infix and Postfix,

Applications of Stack: Reversing Data, Converts Decimal to Binary, Parsing, Postponement, expression Conversion, and evaluation.

Unit III Queue **08 Hours**

Queue: Introduction, Definition, ADT for queue, Storage Methods, Queue Operations, Enqueue, Dequeue, Queue front, Queue rear, Queue Example, Create Queue, priority Queue, Circular Queue.

Application of Queue: Categorising Data, Queue Simulation.

Unit IV Linear Lists **08 Hours**

Introduction, singly linked list, Circularly Linked List, Doubly Linked lists, Basic operations, - Insertion, Deletion, retrieval, traversal, create List, insert node, delete node, List Search, Empty list, Destroy list.

Unit V Linked Stacks and Linked Queues **08 Hours**

Introduction, Operations on Linked stacks and Linked Queues, Dynamic Memory management and Linked Stacks, Implementation of Linked Representations.

Unit VI Overview of Real time Applications of Linear Data Structures **08 Hours**

Stacks – Balancing of Symbols, Infix to Postfix, Evaluation of Postfix expression, Implementing Function Calls, Finding of Spans, undo sequence in text editor, Matching Tags in HTML and XML.
Linked List – Implement Stack using Linked List.
Queues – Scheduling Jobs, Simulation of real-world queues such as ticket counter or first come first served scenarios, Asynchronous Data Transfer.

Textbooks

1. Brassard & Bratley, ―Fundamentals of Algorithmics‖, Prentice Hall India/Pearson Education, ISBN 13-9788120311312.
2. Horowitz and Sahani, ―Fundamentals of Data Structures in C++‖, University Press, ISBN 10: 0716782928 ISBN 13: 9780716782926.
3. Goodrich, Tamassia, Goldwasser, ―Data Structures and Algorithms in C++‖, Wiley publication, ISBN-978-81-265-1260-7
4. Data Structure and Algorithmic Thinking with Python, CareerMonk Publications, Narasimha Karumanchi, 2016

Reference Books

1. Richard F Gilberg & Behrouz A Forouzan, Data Structures (A Pseudocode Approach with C), second edition, Cengage Learning, 2004.
2. PAI, Data Structures, Tata McGraw-Hill Education, 2008
3. Mayank Patel, Data Structure and Algorithm With C, Edu creation Publishing, 2018
4. Thomas H. Cormen, Charles E Leiserson, Ronald L Rivest, Clifford Stein, Introduction to Algorithms, MIT Press, 2001.

List of Assignments

1. Show how you can efficiently implement one stack using two queues.
2. What is the most appropriate data structure lo print elements of queue in reverse order?
3. You are given a pointer to the first element of a linked list L. There are two possibilities for L, it either ends (snake) or its last clement points back to one of the earlier elements in the list (snail). Give an algorithm that tests whether a given list L is a snake or a snail.
4. Check whether the given linked list is either NULL-terminated or ends in a cycle (cyclic).
5. Find nth node from the end of a Linked List
6. Simulate real world queues such as ticket counter.

List of Laboratory Exercises

1. Study assignment on programming IDE Tools
2. Write a program to implement one dimensional array.
3. Write a program to design tic-tac-toe game
4. Write a program to perform basic operation on stack.
5. Write a program to convert and evaluate polish notations.
6. Write a program to perform basic operation on stack.
7. Write a program to implement Priority queue & Double Ended Queue.
8. Write a program to perform basic operation on circular queue.
9. Write a program to implement hashing technique.

10. Write a program to implement searching and sorting techniques

Project Based Learning

1. Expression Evaluation
2. Traffic Management System
3. Library Management System
4. Employee Record System
5. Dictionary
6. Calendar Application
7. Medical Store Management System
8. Cricket Score Sheet
9. Bank Management System
10. Telephone directory

(Note:- *Students in a group of 3 to 4 shall complete any one project from the above list)

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Computational Thinking and Programming Concepts

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture	04 Hours/Week	University Examination	60 Marks	Lecture	04
Practical	02 Hours/Week	Internal Assessment	40 Marks		
		Term Work	50 Marks	Practical	01
		Practical	50 Marks		
		Total	200 Marks	Total	05

Course Objective

The aim of this course is to make students to think in a computational manner to a point where they can derive simple algorithms and code the programs to solve some basic problems in their domain of studies.

Prerequisite:

-

Course Outcomes: On completion of the course, students will have the ability to:

1. Formulate a problem and express its solution in such a way that a computer can effectively carry it out.
2. Apply the Computational Thinking (CT) concepts on case studies/problem-based scenarios through hands-on practice of the CT processes.
3. Write algorithm and pseudo code for the identified strategy
4. Use Abstraction and Modelling.
5. Solve given problems through scratch based graphical programming tool
6. Demonstrate logical and algorithmic thinking.

Unit I Overview of Problem Solving **08 Hours**

Problem Solving Concepts: Formal Problem Definition, Challenges in Problem Solving, Problem solving with Computers, Framework for Problem Solving.
Introduction to Problem solving tools: Flowcharts, algorithm, pseudocode, Data structures.

Unit II Logical and Algorithmic Thinking **08 Hours**

Inductive Vs Deductive arguments, Logic, Boolean Logic, Symbolic Logic, Logical operators and their symbols, Propositional Logic
Algorithmic Thinking: Algorithms, Intuition vs precision, defining algorithms, Algorithm constructs, Controlling algorithm execution, Complex conditionals.

Unit III Overview of Computational Thinking **08 Hours**

About Computational Thinking, Data Representation and Abstraction - Problem formulation, Devising a Solution, Decomposition, Pattern recognition, Generalisation, Evaluation.

Unit IV Overview of Programming Concepts **08 Hours**

Scratch Programming – Working of Scratch, Scratch tool, Motions and Drawing, Looks and Sound, Procedures, Variables, Making decisions, Loops, String Processing, Lists.
Introduction to higher level programming languages like C, Python, C++ and its constructs.

Unit V Limits of Computation **08 Hours**

Capacity Measurement in Computers, Estimate of Physical limitations, Benchmarks, Counting the performance, impractical algorithms, Metaphysical limitations, Impossible algorithms.

Unit VI Computational Thinking in Software Development **08 Hours**

Effective Building Blocks: Basic Algorithms Constructs, Program State, Code Organization, Using Abstractions and Patterns, Effective Modelling: Objectives, Entities, Relationship, Processes, Usage and General Advice.

Testing and Evaluating Programs, Anticipating Bugs, Syntax vs semantic errors, Defensive programming, Verification and validation, Testing the Parts, Testing the Whole, Debugging Case Study: Home Automation System

Textbooks

1. Computational Thinking, By Peter J. Denning and Matti Tedre, The MIT Press Essential Knowledge series
2. Computational Thinking and Coding for Every Student, Jane Krauss, Kiki Prottsman by Corwin Publishers
3. Computational Thinking for the modern problem solver, David D riley, Kenny A Hunt, CRC Press, 2014
4. Computational thinking a beginner's guide to problem solving and programming, Karl Beecher, BCS Learning & Development, 2017

Reference Books

1. How to Solve it by Computer by R. G. Dromey, 1e, Pearson Education.
2. Learn to program with Scratch, Majed Marji, no starch press, 2014
3. Let Us C, Yashavant Kanetkar, Infinity Science Press, 2008
4. Let Us C++, Yashavant Kanetkar, BPB Publications, 1999
5. Introduction to Computation and Programming Using Python, Mit Press, John Guttag, 2016

List of Assignments

1. The Following problems can be solved using SCRATCH Tool: Create a function block that calculates the force needed to accelerate 2,000 kg car 3 m/s2
2. Write different procedures to draw each letter of your name. Name each procedure for the letter that it draws. Then write a script that calls these procedures so you can draw your name on the Stage
3. Write a program that prompts the user to enter five test scores between 1 and 10. The program will then count the number of scores that are greater than 7
4. The Pythagorean theorem states that if a and b are the lengths of the legs of a right triangle and c is the length of the hypotenuse (the longest side), then a2 + b2 = c2 . Write a program that gets three numbers from the user and determines whether they could represent the sides of a right triangle.
5. Create two lists for storing the items sold in a grocery store and their corresponding prices. Write a program that asks the user to enter an item's name and then displays that item's price, if it is found in the list.
6. Write a program that prompts the user to enter the highest and lowest temperatures for the 12 months of a year. Store the input values in two lists.

List of Laboratory Exercises

1. WAP to SWAP (interchange) 2 numbers without using third variable
2. WAP to find the sum and average of values appearing at the positions divisible by 3 in the given sequence of n values
3. WAP that receives any year from the keyboard and uses a function to determine whether the year is a leap year or not.
4. WAP that uses a function that converts a lowercase character to its uppercase
5. WAP to read n numbers and count even and odd numbers.

6. WAP that uses a recursive function to convert given decimal number into its binary equivalent.
7. WAP to use the suitable function to obtain the prime factors recursively.
8. WAP that uses a function that prints the nth element of Fibonacci series using recursion method.
9. WAP that uses a function to calculate the sum of n odd integers.
10. WAP that uses a function power that calculates the power of a given number.

Project Based Learning

1. Identify any patterns in the problem.
2. Build Model for various Mathematical Formulas
3. Study the friendship link of any social networking site.
4. Using primary data source study, the voting pattens of our country.
5. Analyse how algorithms effect social media feeds
6. Visualize and Interpret performance of Athlete for any Sport
7. Modularize a given problem into sub problems.
8. Analyse the next moves of a player for Game of Chess
9. Devise a strategy to compute Result of a particular Class
10. Library Management System

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Programming Technologies and Tools Laboratory – I

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		**Marks**		**Credits**
Lecture	**01 Hours/Week**	**Term Work**	**25 Marks**	**Lecture**	**01**
Practical	**02 Hours/Week**	**Practical**	**25 Marks**	**Practical**	**01**
		Total	**50 Marks**	**Total**	**02**

Course Objective
The course is designed to provide complete knowledge of C language. Students will be able to develop logics which will help them to create programs, applications in C. Also, by learning the basic programming constructs they can easily switch over to any other language in future.

Prerequisite:
-

Course Outcomes: On completion of the course, students will have the ability to:
1. Demonstrate the knowledge of C programming Concepts
2. Develop C programs
3. Define Data types and use them in data processing programs.
4. Trace the execution of programs written in C language
5. Write functions and implement.
6. Analyse and interpret the concept of declarations, initialization, operations on pointers and their usage.

Unit I Basics and Operators **08 Hours**
General problem-solving strategies, Top-down design, Introduction to program Planning tools- algorithm, flowcharts, and pseudo codes. Introduction to Logic Structures: Sequential structure, Decision Structure, Loop Structure. Features of C, basic concepts, structure of C program, program, declarations, variables, data types, expressions, operator's assignment, arithmetic, relational, logical, increment and decrement, precedence of operators, type conversions, scanf and printf functions

Unit II Control structures **08 Hours**
if-else, nested if-else, cascaded if-else and switch statement. C Conditional control structures: for, while do-while Unconditional control structures: break, continue, goto statement.

Unit III Arrays and strings **08 Hours**
Declaration initialization of one-dimensional Array, two-dimensional array, accessing array elements, Character Array/String, Character - Handling Library Functions, Standard Input/Output Library Functions for string.

Unit IV Functions and structures **08 Hours**
What is a Function, Benefits of a Function, Function Terminology, Array of Structures, How does Function Works , Scope and Lifetime of Variables in function ,Storage Classes of Variables , Call by value and call by reference ,Recursion ,Overview of Structures , Defining and Using a Structure , Structures within a Structure

Unit V Pointers **08 Hours**
Declaring and Initializing Pointers, Function and Pointer Parameters, Pointer Arithmetic, Pointer and Arrays, Two Dimensional Arrays and Pointers.

Unit VI Files **08 Hours**

FILE, Opening and Closing of Files, Writing and Reading in Text Format, Writing and Reading in Binary Format, Command Line Arguments

Textbooks

1. Let Us C by Yashavant Kanetkar, 13e, BPB Publication.
2. Brain W.Kernighan & Dennis Ritchie, C Programming Language, 2nd edition, PHI
3. E.Balaguruswamy, Programming in ANSI C 5th Edition McGraw-Hill
4. How to Solve it by Computer by R. G. Dromey, 1e, Pearson Education.

Reference Books

1. C: The Complete Reference by Herbert Schildt.

List of Laboratory Exercises

1. Write a program to read a four-digit integer and print the sum of its digits.
2. Use recursive function calls to evaluate F(x) = x – x3 / 3! + s5/5! –x7/7! + ……
3. WAP to print the table of n.
4. Write a 'C' Program to evaluate Ackerman Function
5. Given a list of marks ranging from 0 to 100, write a program to compute and print the number of students:
   (a) who have obtained more than 80 marks,
   (b) who have obtained more than 60 marks,
   (c) who have obtained more than 40 marks,
   (d) who have obtained 40 or less marks,
   (e) in the range 81 to 100,
   (f) in the range 61 to 80,
   (g) in the range 41 to 60, and
   (h) in the range 0 to 40.
6. Make a Book Shop Inventory. The list should include details such as author, title, price, publisher, stock position. When a particular title and author name is given as input the program should reply whether it is in the list or no. If not, appropriate message should be displayed.
7. Write a program to find the total number of characters in a file.
8. Write a function which takes to integer as argument and return their average in float. WAP to test this function.
9. WAP to read n numbers and count even and odd numbers
10. Write a function which takes to integer as argument and return their sum. WAP to test this function.

Project Based Learning

1. Hangman Game
2. Modern Periodic Table
3. Pacman Game
4. Personal Diary Management System
5. Phonebook Application
6. Quiz Game
7. School Billing System
8. Snake Game
9. Telecom Billing System
10. Tic-Tac-Toe Game
11. Typing Tutor

B.TECH (Computer Engineering)

SEMESTER – II

COURSE SYLLABUS

Mathematics for Computing - II

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	03 Hours/Week	University Examination	60 Marks	Lecture	03
Tutorial:	01 Hours/Week	Internal Assessment	40 Marks	Tutorial	01
		Total	100 Marks	Total	04

Course Objectives:
To equip students with knowledge of:

- Fourier series and integral transforms.
- Multiple integrals and its applications.
- Vector calculus and its applications.

Prerequisite:
The students should have knowledge of vector algebra, derivative, and integration.

Course Outcomes: On completion of the course, students will have the ability to:

1. Use periodic functions as fourier series.
2. Apply methods of finding fourier and Z-transforms.
3. Apply methods of laplace transform of piecewise continuous functions.
4. Identify concepts of double and triple integrals.
5. Apply vector derivative for physical quantities.
6. Evaluate line, surface, and volume integrals.

Unit I Fourier Series **06 Hours**
Definition, Dirichlet's conditions, Fourier Series and Half Range Fourier Series, Harmonic Analysis

Unit II Fourier and Z-Transform **06 Hours**
Fourier Transform (FT): Complex Exponential Form of Fourier series, Fourier Integral Theorem, Sine & Cosine Integrals, Fourier Transform, Fourier Sine and Cosine Transform and their Inverses. Introductory
Z-Transform (ZT): Definition, Standard Properties, ZT of Standard Sequences and their Inverses. Solution of Simple Difference Equations.

Unit III Laplace Transform and its application **06 Hours**
Definition of LT, Inverse LT. Properties & theorems. LT of standard functions. LT of some special functions viz., Periodic, Unit Step, Unit Impulse, ramp, jump, Problems on finding LT & inverse LT. Applications of LT and Inverse LT for solving ordinary differential equations.

Unit IV Multiple Integrals and their Application **06 Hours**
Double and Triple integrations, Applications to Area, Volume, Mean and Root Mean Square Values, moment of inertia, centre of gravity

Unit V Vector Differential Calculus **06 Hours**
Physical interpretation of Vector differentiation, Vector differential operator, Gradient, Divergence and Curl, Directional derivative, Solenoidal, Irrotational and Conservative fields, Scalar potential, Vector identities.

Unit VI Vector Integral Calculus and Applications **06 Hours**
Line, Surface, and Volume integrals, Work-done, Green's Lemma, Gauss's Divergence theorem, Stoke's theorem. Applications to problem in engineering.

Textbooks

1. P. N. Wartikar and J. N. Wartikar, Applied Mathematics (Volumes I and II), 7th Ed., Pune Vidyarthi GrihaPrakashan, Pune, 2013.
2. B. S. Grewal, Higher Engineering Mathematics, 42nd Ed., Khanna Publication, Delhi
3. B.V. Ramana, Higher Engineering Mathematics, 6th Ed., Tata McGraw-Hill, New Delhi, 2008.

Reference Books

1. Erwin Kreyszig, Advanced Engineering Mathematics, 10th Ed., John Wiley & Sons, Inc., 2015.
2. Peter V. O'Neil, Advanced Engineering Mathematics, 7th Ed., Cengage Learning, 2012.

List of Assignments

Six assignments to be given by the course coordinator (Theory)-one from each unit

Project Based Learning - Provisional List of Projects

Students are expected prepare report on any one topic, write its definition, applications and illustrate with few examples. Also, write pseudo code for it, wherever applicable.

1. Fourier series
2. Harmonic analysis
3. Fourier transform
4. Z-Transform
5. Laplace transform technique to solve ODE
6. Multiple Integral to evaluate area and volume
7. Directional derivative
8. Divergence and curl
9. Greens theorem
10. Gauss Divergence Theorem
11. Stokes theorem
12. Unit step function
13. Solenoidal and irrotational fields
14. Simple difference equation
15. Periodic functions

Note: - *Students in a group of 3 to 4 shall complete any one project from the above list)

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Physics for Computing Systems

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		**Marks**		**Credits**
Lecture:	**03 Hours/Week**	**University Examination**	**60 Marks**	**Lecture**	**03**
Practical:	**02 Hours/Week**	**Internal Assessment**	**40 Marks**		
		Term work	**25 Marks**	**Practical**	**01**
		Total	**125 Marks**	**Total**	**04**

Course Objective

To impart knowledge of basic concepts in physics relevant to engineering applications in a broader sense with a view to lay foundation for the Computer Engineering and Science.

Prerequisite:

Basic understanding of physics and calculus.

Course Outcomes: On completion of the course, students will have the ability to:

1. Interpret the properties of charged particles to develop modern instruments such as electron microscopy.
2. Appraise the wave nature of light and apply it to measure stress, pressure, and dimension etc.
3. Summarise the structure and properties of lasers to their performance and intended applications.
4. Classify the optical fibre, understanding the structure, types, and its applications in the field of communication.
5. Solve quantum physics problems to micro level phenomena and solid-state physics
6. Explain mechanical properties of solid matter and connect to applications in the field of engineering.

Unit I Modern Physics **06 Hours**

Motion of a charged particle in electric and magnetic fields, Electrostatic and Magnetostatic focusing, Electron microscope, Wavelength and resolution, Specimen limitation, Depth of field and focus, Transmission electron microscope (TEM), Scanning electron microscope (SEM), Separation of isotopes by Bainbridge mass spectrograph, Cathode ray tube (CRT).

Unit II Wave Optics **06 Hours**

Interference: Interference of waves, interference due to thin film (Uniform and nonuniform (only formula-no derivation is expected), Newton's ring, Applications of interference (optical flatness, highly reflecting films, non-reflecting coatings).

Diffraction: Introduction, Classes of diffraction, Diffraction at a single slit (Geometrical method), Conditions for maximum and minimum, Plane diffraction grating, Conditions for principal maxima and minima

Polarisation: Introduction, Double refraction and Huygen's theory, Positive and negative crystals, Nicol prism, Dichroism.

Unit III Lasers **06 Hours**

Principle of laser, Einstein's coefficients, Spontaneous and stimulated emission, Population inversion, Ruby laser, Helium-Neon laser, Semiconductor laser, Single Hetro-junction laser, Gas laser: CO2 laser, Properties of lasers, Laser speckles, Applications of lasers (Engineering/ industry, medicine, Computers)

Unit IV Fibre Optic **06 Hours**

Principle of fibre optics, Construction, Numerical Aperture for step index fibre; critical angle, angle of acceptance, V number, number of modes of propagation, types of optical fibres, Fibre optic communication system, advantages, and disadvantages of fibre optics.

Unit V Quantum Mechanics **06 Hours**

Dual nature of matter, DeBroglie's hypothesis, Heisenberg's uncertainty principle with illustrations, Physical significance of wave function, Schrodinger's time dependant and time independent wave equation, Application of Schrodinger's time independent wave equation to the problems of Particle in a rigid box, step potential and potential barrier (analytical discussion), tunnelling effect.

Unit VI Solid state physics **06 Hours**

Free electron theory, Density of states, Bloch theorem (Statement only), Origin of band gap, Energy bands in solids, Effective mass of electron, Fermi-Dirac probability function and position of Fermi level in intrinsic semi-conductors (with derivation) and in extrinsic semi-conductors, Band structure of p-n junction diode under forward and reverse biasing, Conductivity in conductor and semi-conductor, Hall effect and Hall coefficient, Photovoltaic effect, Solar cell, and its characteristics.

Textbooks

1. A Textbook of Engineering Physics, M N Avadhanulu, P G Kshirsagar and TVS Arun Murthy, S. Chand Publishing (2018)
2. Engineering Physics, R K Gaur and S L Gupta, Dhanpat Rai Publishing Co Pvt Ltd (2015)
3. Concepts of Modern Physics, Arthur Beiser, Shobhit Mahajan and S. Rai Choudhury, McGraw Hill Education (2017)

Reference Books

1. Fundamentals of Physics, Jearl Walker, David Halliday and Robert Resnick, John Wiley and Sons (2013
2. Optics, Francis Jenkins and Harvey White, Tata Mcgraw Hill (2017)
3. Principles of Physics, John W. Jewett, Cengage publishing (2013
4. Introduction to Solid State Physics, C. Kittel, Wiley and Sons (2004)
5. Principles of Solid-State Physics, H. V. Keer, New Age International (1993)
6. Laser and Non-Linear Optics, B. B. Laud, New Age International Private Limited (2011)
7. Nanotechnology: Principles and Practices, Dr. S. K. Kulkarni, Capital Publishing Company (2014)
8. Science of Engineering Materials- C.M. Srivastava and C. Srinivasan, New Age International Pvt. Ltd. (1997)
9. Introduction to Electrodynamics –David R. Griffiths, Pearson (2013
10. Renewable Energy: Power for a Sustainable Future, Boyle, Oxford University Press (2012)

List of Assignments

Six assignments to be given by the course coordinator (Theory)-one from each unit/one mini project with report-students can work in group of 4 Maximum.

List of Laboratory Exercises

1. Study of lissajous figure by Cathode Ray Oscilloscope (CRO)
2. Determination of e/m by Thomson method
3. Determination of radius of planoconvex lens/wavelength of light/Flatness testing by Newton’s rings
4. Determination of wavelength of light using diffraction grating
5. Determination of resolving power of telescope
6. Determination of thickness of a thin wire by air wedge
7. Determination of refractive index for O-ray and E-ray
8. Determination of divergence of a laser beam
9. Particle size by semiconductor laser
10. Determination of wavelength of laser by diffraction grating
11. To study Hall effect and determine the Hall voltage
12. Calculation of conductivity by four probe methods
13. Study of solar cell characteristics and calculation of fill factor
14. Determination of band gap of semiconductor
15. Determination of Planck's Constant by photoelectric effect

Project Based Learning - Provisional List of Projects

1. Measurement and effect of environmental noise in the college
2. Design and simulation of automatic solar powered time regulated water pumping
3. Solar technology: an alternative source of energy for national development
4. Design and construction of digital distance measuring instrument
5. Design and construction of automatic bell ringer
6. Design and construction of remote-control fan
7. Design and construction of sound or clap activated alarm
8. Electronic eye (Laser Security) as auto switch/security system
9. Electric power generation by road power
10. Determination of absorption coefficient of sound absorbing materials
11. Determination of velocity of O-ray and E-ray in different double refracting materials
12. Need of medium for propagation of sound wave
13. Tesla Coil
14. Thin film interference in soap film-formation of colours
15. LiFi- wireless data transfer system using light

(Note: - *Students in a group of 3 to 4 shall complete any one project from the above list)

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Numerical Computation

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		**Marks**		**Credits**
Lecture:	**04 Hours/Week**	**University Examination**	**60 Marks**	**Lecture**	**04**
		Internal Assessment	**40 Marks**		
		Total	**100 Marks**	**Total**	**04**

Course Objective

To equip students with the knowledge of:

- Numerical methods to solve linear and system of linear equations.
- Numerical methods for differentiation and integrations.
- Numerical methods for ordinary and partial differential equations

Prerequisite:

Mathematics for Computing - I

Course Outcomes: On completion of the course, students will have the ability to:

1. Apply methods to solve linear and transcendental equations.
2. Solve system of linear equations.
3. Compute finite differences.
4. Apply method for numerical differentiation and integration.
5. Solve ordinary differential equations numerically.
6. Apply methods to solve partial differential equations.

Unit I Solution of Algebraic and Transcendental Equation **08 Hours**

Bisection method, Method of false position, Newton's method and Newton-Raphson method, Approximate solution of equation – Horner's method

Unit II Solution of Linear Simultaneous Equation **08 Hours**

Gauss elimination method, Gauss-Jordan method, Crout's triangular method, Iterative method of solution- Jacobi iteration method, Gauss-Seidal iteration method, Relaxation method

Unit III Finite Differences **08 Hours**

Forward difference operator, Backward difference operator, Central difference operator, Newton's interpolation formula, Newton's forward–backward-central interpolation formula, Sterling formula, Bessel's formula, Interpolation with unequal intervals.

Unit IV Differentiation and Integration **08 Hours**

Newton-Cortes's formula, Trapezoidal rule, Simpson one–third rule, Simpson three- eighth rule, Weddle's rule.

Unit V Numerical Solution of ODE **08 Hours**

Picard's methods, Taylor series method, Euler's method, Modified Euler's method, Runge – Kutta method, Predictor–corrector method, Milne's method. Adams-Bash fourth method, Second–order differential equation

Unit VI Finite Difference Methods **08 Hours**

Finite difference methods for solving second order two - point linear boundary value problems - Finite difference techniques for the solution of two-dimensional Laplace's and Poisson's equations on rectangular domain – One dimensional heat flow equation by explicit and implicit (Crank Nicholson) methods – One dimensional wave equation by explicit method

Textbooks

1. Burden, R.L and Faires, J.D, "Numerical Analysis", 9th Edition, Cengage Learning, 2016.
2. Grewal, B.S., and Grewal, J.S., "Numerical Methods in Engineering and Science", Khanna Publishers, 10th Edition, New Delhi, 2015.
3. S. S. Shastri, Introduction to Numerical Methods, PHI Publication.
4. V. Rajaraman, Computer Oriented Methods, 3rd Edition, PHI Publication.

Reference Books

1. Steven C Chapra, Numerical Methods for Engineers, 5th Edition, McGraw Hill Publication
2. James F. Epperson, An Introduction to Numerical Methods and Analysis, 2nd Edition, Wiley Publication.

List of Assignments

Six assignments to be given by the course coordinator (Theory)-one from each unit.

Project Based Learning - Provisional List of Projects

Students are expected prepare report on any one topic, write its definition, applications and illustrate with few examples. Also, write pseudo code for it, wherever applicable.

1. Bisection method
2. Newton Raphson's method
3. Horner's method
4. Crouts triangular mehod
5. Gauss Seidel method
6. Jacobi Method
7. Interpolation
8. Trapezoidal Rule
9. Simpson's rules
10. Euler method
11. Runge kutta method
12. Finite difference technique
13. Crank Nicolson method
14. Predictor Corrector method
15. Relaxation method

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Electrical Technology

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	04 Hours/Week	University Examination	60 Marks	Lecture	04
Practical:	02 Hours/Week	Internal Assessment	40 Marks		
		Term work	25 Marks	Practical	01
		Total	125 Marks	Total	05

Course Objective

To equip students with the knowledge of power system basics, magnetic circuits electrical machines, transformers, wiring, measurements, illumination, and batteries.

Prerequisite:

-

Course Outcomes: On completion of the course, students will have the ability to:

1. Explain the various parameters related to magnetic circuit.
2. Describe basic concepts of AC fundamentals and circuits.
3. Illustrate constructional features and describe different parameters of transformer.
4. Describe basic concepts of power system and three phase circuits.
5. Demonstrate AC and DC electrical machines.
6. Classify types of batteries.

Unit I Magnetic Circuits **08 Hours**

Magnetic effect of electric current, Cross & Dot Convention, Right hand thumb rule, Concept of flux, flux linkages, magnetic field, magnetic field strength, magnetic field intensity, absolute permeability, relative permeability Kirchhoff's laws for magnetic circuits. Magnetic circuit concepts, analogy between electric & magnetic circuits, magnetic circuits with DC and AC excitations, magnetic leakage, B-H curve, hysteresis and eddy current losses, magnetic circuit calculations, mutual coupling.

Unit II AC Fundamentals and circuits: AC Fundamentals **08 Hours**

Sinusoidal, square, and triangular waveforms – average and effective values, form and peak factors, concept of phasor, phasor representation of sinusoidally varying voltage and current. Analysis of series, parallel and series parallel RLC Circuits: apparent, active & reactive powers, power factor, causes and problems of low power factor, power factor improvement; resonance in series and parallel circuits, bandwidth, and quality factor (simple numerical problems.

Unit III Single Phase Transformer **08 Hours**

Faradays law of electromagnetic induction, statically and dynamically induced emf, self-inductance, mutual inductance, coefficient of coupling. Single Phase Transformer: Principle of operation, construction, e.m.f. equation, voltage ratio, current ratio, KVA rating, determination of efficiency and regulation by direct load test, equivalent circuit, power losses, (simple numerical problems), introduction to auto transformer. Three phase transformer and its different winding connections.

Unit IV Introduction to Power System and Three Phase **08 Hours**

Circuits: General layout of electrical power system and functions of its elements, standard transmission and distribution voltages, concept of grid (elementary treatment only) Power generation to distribution through overhead lines and underground cables with single line diagram. Three phase system-its necessity and advantages, meaning of phase sequence, star and delta connections, balanced supply and balanced load, line, and phase voltage/current relations, three phase power and its measurement (simple numerical problems).

Unit V Electrical Machines **08 Hours**

DC & AC: Principles of electromechanical energy conversion, DC machines: types, e. m. f. equation of generator and torque equation of motor, characteristics, and applications of dc motors (simple numerical problems). single Phase Induction motor: Principle of operation and introduction to methods of starting, applications. Three Phase Induction Motor: types, Principle of operation, slip-torque characteristics, applications (numerical problems related to slip only).

Unit VI Batteries **08 Hours**

Basic idea of primary and secondary cells, Construction, working principle and applications of Lead-Acid, Nickel Cadmium and Silver-Oxide batteries, charging methods used for lead-acid battery (accumulator), Care and maintenance of lead-acid battery, Series and parallel connections of batteries, General idea of solar cells, solar panels and their applications, Introduction to maintenance free batteries, Safe disposal of Batteries; Fuel cell: Principle & Types of fuel cell.

Textbooks

1. B.L.Theraja, A Textbook of Electrical Technology, Vol.1, S.Chand& Company Ltd. New Delhi
2. V.K.Mehta, Basic Electrical Engineering, S Chand & Company Ltd. New Delhi.
3. J.Nagarath and Kothari, Theroy and applications of Basic Electrical Engineering, Prentice Hall of India Pvt. Ltd.

Reference Books

1. Electrical Technology - Edward Huges (Pearson
2. Basic Electrical Engineering - D. P. Kothari, J Nagarath (TMC)
3. Electrical power system technology - S. W. Fordo, D. R. Patric (Prentice Hall)
4. Electrical, Electronics Measurements and Instruments - (Satya Prakashan)

List of Assignments

Six assignments to be given by the course coordinator (Theory)-one from each unit.

List of Laboratory Exercises

1. Plotting B-H characteristics for a material.
2. Load test on single phase transformer.
3. Testing and maintenance of batteries.
4. Verification of voltage and current relationships in star and delta connected 3-phase networks.
5. Load test on DC machine.
6. To find the performance of series R-L-C circuit at different condition
7. OS & SC test on single phase transformer to find efficiency and regulation
8. Speed control of DC motor
9. Study of different types of starters for DC & AC Machine
10. Load test on 3 phase Induction moto

Project Based Learning – Provisional List of Projects

1. Building a small resistive load lamp bank
2. Building a small resistive load lamp bank for various types of connections like series, parallel, star, delta
3. Building a small inductive load lamp bank for various types of connections like series, parallel, star, delta
4. Building a small capacitive load lamp bank for various types of connections like series, parallel, star, delta
5. Building a small resistive load lamp bank
6. Building a staircase wiring model on a board
7. Building a Go down wiring model on a board
8. Rewinding of a choke
9. Rewinding of a small transformer
10. Building a small rectifier circuit on bread board
11. Building a mobile charger circuit on a bread board
12. Building an electric buzzer circuit
13. Building a solar charger for mobile phone
14. Building a small wind turbine
15. Small Agricultural pump model with DC motor
16. Small Agricultural pump model with AC motor

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Paradigms of Programming

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	04 Hours/Week	University Examination	60 Marks	Lecture	04
Practical:	02 Hours/Week	Internal Assessment	40 Marks		
		Term Work & Practical	50 Marks	Practical	01
		Total	150 Marks	Total	05

Course Objective

The course aim to make students aware of various programming paradigms and emphasising on using object-oriented approach to solve real world problems.

Prerequisite:

-

Course Outcomes: On completion of the course, students will have the ability to:

1. Demonstrate the knowledge of different programming paradigms.
2. Demonstrate the concepts of Object-Oriented Paradigm.
3. Develop programs using object-oriented approach
4. Develop small size programs using different programming language and Paradigm
5. Compare the strengths and weakness of different programming language specific to application context
6. Recognize the concepts of same kind from different programming languages and paradigms

Unit I Overview of programming paradigms **08 Hours**

Basic elements of programming languages, compiled vs. interpreted, syntax, semantics, data types, Imperative languages and non-imperative, Scripting languages, Data-oriented languages, Object-oriented languages, Event-driven Programming

Unit II Functional Programming **08 Hours**

Definition of a function: domain and range, total and partial functions, strict functions, Recursion, Referential transparency

Unit III Logic Programming **08 Hours**

Basic constructs, Facts, rules, queries, processing, goals, predicates, variables, existential queries, conjunctive queries, Definition, and semantics of a logic program.

Recursive programming: Computational model of logic programming

Unit IV Object Oriented programming **08 Hours**

Basic concepts: objects, classes, methods, overloading methods, messages inheritance: overriding methods, single inheritance, multiple.

inheritance Interfaces (e.g., in Java), encapsulation, polymorphism.

Unit V Overview of Languages **08 Hours**

Ruby: basic concepts, interpreter, strings, control structures, conditionals, loops, (duck) typing, arrays, hashes, symbols.

Prolog: structures, matching structures, equality, comparison operators, arithmetic's, lists, splitting lists, enumerating lists.

Unit VI Advanced Programming **08 Hours**

Concurrent programming, serial vs. parallel programming, process communication, basic concepts, data types, atoms, variables, pattern matching, lists, tuples- Database Programming, ,Internet programming design principles, windows programming.

Textbooks

1. Seven Languages in Seven Weeks, Bruce A. Tate, Pragmatic Bookshelf, 2010
2. Programming Languages: Principles and Paradigms, Maurizio Gabrielli, Simone Martini, Springer, 2010
3. Programming Languages - Principles and Paradigms, Allen B. Tucker, Robert E. Noonan: (2nd ed.) McGraw-Hill, 2007
4. Clark R. G., Comparative Programming Languages, Addison-Wesley (3rd Ed.), 2000.
5. Mitchell, J. C. Concepts in Programming Languages, Cambridge University Press, 2002
6. Sebesta, R. W., Concepts of Programming Languages, Global Edition, Addison-Wesley (11th Ed.) 2016
7. Programming Languages: Concepts and Constructs; 2nd Edition, Ravi Sethi, Pearson Education Asia, 1996.

Reference Books

1. Programming Language Principles and Practice by KC Louden
2. Language manuals and on-line resources for programming languages, tools, and projects.

List of Assignments

Six assignments to be given by the course coordinator (Theory)-one from each unit

List of Laboratory Exercises

1. Write a Simple Program (as given by course coordinator) in Ruby
2. Write a simple Program (as given by course coordinator) in Prolog
3. Write a program to Implement Concept of Class and Objects.
4. Write a Program to Implement Concept of Method Overloading and Method Overriding
5. Write a program to implement Concept of Inheritance.
6. Write a program to implement Concept of Interface.
7. Write a program to implement Concept of Recursive Function.
8. Study of Database Programming Language approach.

Project Based Learning - Provisional List of Projects

Use the best programming paradigm for the following:

1. Operations on Matrix
2. Recursion
3. Referential transparency
4. The countdown problem
5. tic-tac-toe
6. Lazy evaluation strategy

7. Assume that you have a list of temperature readings from several cities in the world. Some of them are in Celsius and some in Fahrenheit. First let us convert them all to Celsius, then let us print the data neatly.
8. Implement a better password protection scheme: In the program {User, Password} pairs are sent in plain text over the net. Implement a scheme where the password is never stored, instead store the MD5 checksum of the password and transmit this over the net.
9. All users have the same rights: Implement a scheme whereby different users are restricted to which directories they may access.
10. Files are sent as atomic actions: Files are read, transmitted, and written as atomic actions. This may not work if the files become very large. Implement a scheme for sending the files in smaller chunks. Implement a scheme whereby an FTP transfer can be aborted and restarted in the case where we transfer very large files.

Programming Technologies and Tools Laboratory – II

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		**Marks**		**Credits**
Lecture	**01 Hours/Week**	**Term Work**	**25 Marks**	**Lecture**	**01**
Practical	**02 Hours/Week**	**Practical**	**25 Marks**	**Practical**	**01**
		Total	**50 Marks**	**Total**	**02**

Course Objective
The course focuses on making students learn and practise the Object-Oriented programming, to use concepts and solve the problems.

Prerequisite:
-

Course Outcomes: On completion of the course, students will have the ability to:
1. Choose and apply different Concepts of OOP
2. Demonstrate the use of functions to solve real world problem
3. Identify and apply the concept of Access Specifiers, Scope Resolution operator, Data Abstraction
4. Compare different types of inheritance to solve given problem.
5. Develop applications with constructor and polymorphism.
6. Develop OOP applications using file Handling and Exception handling.

Unit I Introduction to Object Oriented Programming with C++ **08 Hours**
Introduction to Object Oriented Programming, Basic Concept of OOP, Need for OOP, Benefits of OOP, Object Oriented Languages, Applications of OOP. C verses C++, C++ Characteristics, Structure of C++ program, Tokens, Keywords, Identifiers and Constants, Data Types, Declaration of variables, Dynamic initialization of variables, Control Structures

Unit II Functions in C++ **08 Hours**
The Main Function, Function Prototyping, Call by Reference, Inline functions, Default arguments, Function Overloading, Operator Overloading, Operator precedence, Math library functions, Friend and Virtual Functions.

Unit III Classes and Data Abstraction **08 Hours**
Class specification, Class Objects, Scope resolution operator, Access specifiers Public, Private, Protected, Defining member Functions, Nesting of Member Functions, Private Member Functions, Static Data Members, Static Member Functions, Data hiding.

Unit IV Inheritance **08 Hours**
Defining Derived Classes, Types of Inheritance, Virtual Base Class, Abstract class. Inheritance and protected members, protected base class inheritance, Inheriting multiple base classes, Templates: Class template, class template with parameter, function template, function template with parameter.

Unit V Constructor and destructor **08 Hours**
Types of Constructors, Types of copy constructor, constructor overloading, constructor with default parameter, dynamic initialisation of objects, destructor.
Polymorphism: Base class, Virtual Functions, Pure Virtual Functions, Calling a virtual function through a base class reference, Early and Late Binding.

Unit VI Managing, I/O and Working with Files **08 Hours**

C++ stream classes, Unformatted IO operations, formatted IO operations, Classes for file stream operations, opening and closing files, Different File Operations in C, Exception handling in C++.

Textbooks

1. E. Balaguruswamy, "Object Oriented Programming using C++", 4th Edition,Mc Graw Hill, 2010.
2. The C++ Programming Language, Bjarne Stroustrup.

Reference Books

1. Yashwant Kenetkar," Let us C++",1stEd., Oxford University Press (2006)
2. Bjrane Stroustrup, "C++ Programming language",3rd edition, Pearson education Asia (1997)

List of Laboratory Exercises

1. Explain basic concept of OOP, characteristics of OOP, Difference between C and CPP.
2. Demonstrate Basic simple CPP Program and Program related Control structures in CPP.
3. Demonstrate Concept of Function in CPP.
4. Demonstrate Concept of Inline Function in CPP.
5. Demonstrate Concept of Function Overloading and Operator Overloading in CPP.
6. Demonstrate Concept of Class and Object with the help of Scope Resolution Operator in CPP.
7. Demonstrate Concept of Different types of inheritance in CPP.
8. Demonstrate Concept of Constructor and Destructor in CPP
9. Demonstrate Concept Friend and Virtual Function in CPP
10. Demonstrate Concept of File handling and Exception handling in CPP.

Project Based Learning - Provisional List of Projects

1. Billing Application
2. Traffic Management System
3. Library Management System
4. Employee Record System
5. Security System
6. Calendar Application
7. Medical Store Management System
8. Cricket Score Sheet
9. Bank Management System
10. Telecom Billing System

Computer System Workshop Technology

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Practical:	02 Hours/Week	Term Work	50 Marks	Practical	01
		Practical	50 Marks		
		Total	100 Marks	Total	01

Course Objective

This course focuses on enabling students to identify the hardware components of computer, assembling them, running diagnosis, carry out system configurations and installing system and user applications necessary for computing courses.

Prerequisite

-

Course Outcomes: On completion of the course, students will have the ability to:

1. Identify the architecture of a computer and its different components, including their technology evolution.
2. Apply their knowledge about computer peripherals to identify problems.
3. Install and uninstall given software step-by-step
4. Demonstrate the working of Internet
5. Prepare document using Latex
6. Use GitHub tool for coding and collaboration.

Unit I Computer Hardware **06 Hours**

Introduction to hardware components, random access memory (RAM), Types of RAM & their speed, tips for buying RAM, how to add memory to a computer, problems when installing memory, Central Processing Unit (CPU), Types Of CPU: considerations when buying a new CPU (Types & Differences), different speeds available for CPU and what do they mean, 32 Bit vs 64 Bit – Which One to Choose & Why? How to choose a CPU type for different needs? Graphic Card & Types, how to install a Graphics Card, installing a CD or DVD burner, Jumper Switch settings, Hard Disk upgrade, Different ports and why we use them - USB, PS2, DivX, Graphic card & types, Virtual Memory and how to configure it for optimum system performance.

Unit II Assembly of Computer and Software Installations **06 Hours**

Assembling the motherboard, replacing fan, how to avoid common mistakes during assembly, Installation of system software: Operating system (Windows and Linux), Installation's step for operating system, Dual booting, Configure the BIOS, Installation of Antivirus, Installation of the open-source software such as Scilab, Latex, Installation of Ms Office.

Unit III Basic Diagnostic of Hardware and Software **06 Hours**

Diagnosis of Power Up problem, Boot Drive, Errant Keyboard, mouse problems, slow computer performance, Computer freezes and displays BSOD (Blue screen of death), no display on monitor, no sound, computer rebooting or turning itself off, how to troubleshoot a computer that does not boot, Registry Cleaner.

Unit IV Computer Network Environments **06 Hours**

Network connecting devices. Configure the TCP/IP setting, connect to Local Area Network and access the Internet, Configuring Wireless network. Server and Its Configuration, Email Clients, Browsers, Office tools, customize web browsers with the LAN proxy settings, bookmarks, search toolbars and pop-up blockers, Browsing netiquettes and cyber laws. Cloud Access Tools.

Unit V Configuration of External devices **06 Hours**

Physical set-up of Printers- Performing test print out, Printing of document etc, Scanner set-up, Webcam, Bluetooth device, Memory card reader, Connecting and Using Projectors.

Unit VI Productivity Tools **06 Hours**

Open-Source Tools Such as Latex, GitHub, Latex: Format words, lines, and paragraphs, design pages, create lists, tables, references, and figures in LATEX. Introduction to LaTex Packages and classes. Using Git, Version Control Systems, interacting with GitHub, Reverting Changes, Creating Pull Requests.

Textbooks

1. Introduction to Information Technology, ITL Education Solutions limited, Pearson Education.
2. PC Hardware and A+Handbook – Kate J. Chase PHI (Microsoft)
3. LaTeX Companion – Leslie Lamport, PHI/Pearson
4. Scilab, from theory to practice Scilab: I. Fundamentals Perrine Mathieu, Philippe Roux 2016
5. ISBN: 978-2-8227-0293-5

Reference Books

1. Computer Fundamentals, MS Office, and Internet & Web Technology by Dinesh Maidasani.
2. IT Essentials PC Hardware and Software Companion Guide Third Edition by David Anfinson and Ken Quamme. – CISCO Press, Pearson Education.
3. https://nptel.ac.in/courses/106/105/106105081/
4. http://nptel.ac.in/courses/106105084/

List of Laboratory Exercises

1. Demonstrate the Computer Hardware Components and explain its working.
2. Demonstrate the Networking Components and explain its working.
3. Installation of operating system MS windows, Unix on the personal computer
4. Installation of Application software Scilab, Latex, MS office on the personal computer
5. Troubleshooting hardware related problem.
6. Customize web browsers with the LAN proxy settings, bookmarks, search toolbars and pop-up blockers. Also, plug-ins like Macromedia Flash and JRE for applets should be configured.
7. Execution of Important "layout" and formatting commands in Latex,
8. Installation of Antivirus and customize the browsers to block pop ups, block active x downloads to avoid viruses and/or worms
9. Using Scilab commands, perform basic arithmetic and matrix operations
10. Create a Scilab script file to dis-play product of a matrix A and inverse of A.

Project Based Learning - Provisional List of Projects

1. Collect specifications of similar types of hardware and software and prepare report comparing them
2. Assembling and disassembling the PC back to working condition.
3. Installation of operating systems LINUX on Server and different packages on a PC.

4. Practice hardware troubleshooting exercises related to various components of computer like monitor, drives, memory devices, printers etc. and software troubleshooting related to BIOS etc
5. To start your own computer repair workshop. What would your initial planning involve? What would you look for in terms of building, furnishings, tools and any other equipment that you can think of?
6. Cyber Hygiene: Installing antivirus for Windows.
7. Prepare the report of need of programming language in 21st century.
8. Collect various types of computer hardware and prepare summary report
9. Prepare Seminar report using LaTex
10. Prepare Project report using LaText

B.TECH (Computer Engineering)

SEMESTER – III

COURSE SYLLABUS

Discrete Mathematics and Applications

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		**Marks**		**Credits**
Lecture:	**03 Hours/Week**	**University Examination**	**60 Marks**	**Lecture**	**03**
Tutorial:	**01 Hours/Week**	**Internal Assessment**	**40 Marks**	**Tutorial**	**01**
		Total	**100 Marks**	**Total**	**04**

Course Objective

The courses emphasise on mathematical foundation required for computing enabling the students to develop logical thinking, reasoning, and problem-solving skills.

Prerequisite:

Elementary Linear Algebra, Mathematics for Computing-I

Course Outcomes: On completion of the course, students will have the ability to:

1. Demonstrate the ability to write and evaluate a proof technique.
2. Apply the basic principles of set theory to analyse the data relationship and prove basic properties of set.
3. Demonstrate an understanding of relations and functions to determine their properties.
4. Apply the knowledge of Boolean algebra for building basic electronic and digital circuits.
5. Solve problems of combinatorics and recurrence relations.
6. Model problems in Computer Science using graphs and trees.

Unit I Mathematical Logic **06 Hours**

Propositional Logic, Predicate logic, First order logic, Rules of inference, Introduction to proof techniques, resolution, Mathematical induction, Methods of proofs, Applications.

Unit II Set Theory **06 Hours**

Types of sets, Sets operations and laws, Algebra of Sets, Multisets, Application of the principle of inclusion and exclusion.

Number Theory: Modular arithmetic, prime numbers, and properties, GCD, Chinese remainder theorem, Solving congruences, Applications of congruences Cryptography.

Unit III Relations **06 Hours**

Basic definition, properties and types of relations, relations and digraphs, paths in relations and digraphs, equivalence and partially ordered relations, Transitive closure and Warshall's algorithm.

Functions: Types of functions, Identity functions, Composition of functions, Mathematical functions, Pigeonhole principle.

Unit IV Algebraic Structures **06 Hours**

Isomorphism and Homomorphism Groups, Algebraic Structures with Binary Operations, rings, Cyclic groups, codes.

Lattice: Posets and Hasse Diagrams, Lattice as an algebraic system, Properties of lattices.

Group Codes: The Communication Model and Basic notion of Error Correction, Generation of Codes, Parity Checks, Error recovery in group codes.

Unit V Combinatorics and Recurrence Relations **06 Hours**

Combinatorics: Permutations, Sum rule, Product rule, Combinatorial proofs.

Recurrence Relations: Linear Recurrence relation, Second order recurrence relations with constant coefficients, Applications of Recurrence relation.

Unit VI Graph Theory and Application **06 Hours**

Definition, Degree, Types, Paths, Circuits, Operations on graphs, and Graph Models, Isomorphism, Connectedness, Planar graphs and their properties, Eulerian and Hamiltonian graphs.

Trees: Basic properties of trees, Binary trees, Application: Graph and Networks: Minimum Spanning Tree, Shortest Path, Huffman coding.

Textbooks

1. J.P. Tremblay and Manohar: Discrete mathematical structures with application to Computer Science, McGraw hill- New Delhi.
2. B. Kolman and R.C. Busby: Discrete mathematical structures for computer science Prentice Hall, New-Delhi.
3. S. Malik and M. K. Sen Discrete Mathematics, Cengage Learning India Pvt. Ltd.

Reference Books

1. Kenneth H. Rosen, Discrete Mathematics, and its applications Eighth Edition McGraw Hill Education
2. Stanat and McAlister, Discrete Mathematics for Computer Science, PHI
3. R.M. Somasundaram Discrete Mathematical Structures, Prentice Hall India Learning Private Limited

List of Assignments

1. Given a fact or a statement prove or disprove using suitable technique.
2. Write the given English language sentences represent in the Symbolic logic
3. Given the statement forms Infer the validity of the statement form
4. Draw a Hasse diagram and find chains and antichains
5. Find the number of ways for any event or given sample space.
6. Given a problem represent in a graph and compute the optimal solution
7. Given a communication network find the path between the given nodes

Project Based Learning - Provisional List of Projects

1. Discrete Mathematics in Railway Planning using graph theory and linear algebra.
2. Object transformations using linear algebra.
3. Discrete mathematics in cryptography.
4. In Google maps to determine fastest driving routes and times.
5. In image processing
6. In relation database using sets.
7. In cyber security using graph theory.
8. Shortest path between two cities using a transportation system.
9. Data compression system with the help of Huffman coding.
10. Find the shortest tour that visits each of a group of cities only once and then ends in the starting city using graphs.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Data Structures and Algorithmic Thinking

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
	Hours/Week		Marks		Credits
Lecture:	4 Hours/Week	University Examination	60 Marks	Theory	04
Practical:	2 Hours/Week			Practical	01
		Internal Assessment	40 Marks		
		Term Work	25 Marks		
		Practical	25 Marks		
		Total	150 Marks	Total	05

Course Overview

The course enables students to perform tasks that facilitates them to understand interaction between the algorithms and the structure of the data being analysed by these algorithms. This course also focuses to train students in process of algorithmic thinking enabling them to build simpler solutions to various computational problems.

Prerequisite:

Classical Data Structure, Computational Thinking and Programming Concepts, Programming Technologies, and Tools Laboratory 3

Course Outcomes: On completion of the course, students will have the ability to:

1. Demonstrate the use of ADTs,
2. Develop code to illustrate sorting and searching algorithms.
3. Comprehend the real time problem.
4. Practise and apply Iterative Thinking
5. Practise and apply Recursive Thinking
6. Apply algorithms and data structures in various real-life software problems

Unit I: Graphs Algorithm **08 Hours**

Introduction to Graphs, Application of Graphs, Graph Representation, Graph Traversals, Topological Sort, Shortest Path Algorithms, Minimal Spanning Tree, Graph Algorithm problems & Solutions.

Unit II: Tree **08 Hours**

Introduction to Trees, Binary Trees, Types of Binary Trees, Properties of Binary Trees, Binary Tree Traversals, Generic Trees (N-ary Trees), Threaded Binary Tree Traversals (Stack or Queue-less Traversals), Expression Trees, XOR Trees, Binary Search Tree, Balanced Binary Search Trees, Adelson-Velskii and Landis (AVL) Trees.

Unit III Priority Queues & Heaps **08 Hours**

Introduction to Priority Queues, Priority Queue ADT, Priority Queue Applications, Priority Queue Implementations, Heaps and Binary Heaps, Heapsort, Priority Queue problems and Solutions.

Disjoint Sets ADT – Introduction, Equivalence Relations and Equivalence Classes, Disjoint Sets ADT, Trade-off in Disjoint Sets ADT implementations, Fast UNION Implementation – Slow Find and Quick Find

Unit IV Recursion and Backtracking **08 Hours**

Introduction to recursion, why recursion, Format of a Recursive Function, Recursion and Memory, Recursion Vs. Iteration, Algorithms of Recursion, Recursion problems and Solutions.

Introduction to Backtracking, Algorithms of Backtracking, Backtracking problems, and Solutions.

Unit V Sorting and Searching **08 Hours**

Introduction to Sorting, Classification of Sorting algorithms, Bubble Sort, Selection Sort, Insertion Sort, Shell Sort, Merge Sort, Heap Sort, Quick Sort, Tree Sort, Comparison of Sorting algorithms, Linear Sorting Algorithms, Counting Sort, Bucket Sort, Radix Sort, Topological Sort, External sorting.

Introduction to Searching, Types of Searching, Unordered searching, ordered linear search, Binary search, Interpolation search, Comparison of searching algorithm.

Unit VI Maps, Hash tables and Skip Lists **08 Hours**

Introduction - Hashing, Hash Tables – Hash Functions, Collision Handling schemes, Load Factors, Rehashing, and Efficiency.

Map ADT Introduction, Counting Word Frequencies, Sorted Maps, Sorted Search Tables, Skip Lists – Search and Update Operations in a skip list.

Textbooks

1. Data Structures: A Pseudo code approach with C, R. Gillberg, B. Forouzn
2. Data structures using C and C++ by Langsam, Augenstein, Tenenbaum,PHI publication
3. Data Structure and Algorithmic Thinking with Python, CareerMonk Publications, Narasimha Karumanchi, 2016

Reference Books

1. Data Structures and Algorithms in Python, Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, John Wiley & Sons, 2013
2. Think Data Structures- Algorithms and Information Retrieval in Java, Allen B. Downey, O'Reilly, 2017

List of Assignments

1. Write algorithm, pseudocode, and code to solve Recursion Problems like – Towers of Hanoi, whether given array is in sorted order.
2. Write algorithm, pseudocode, and code to solve Backtracking Problems like - Generate all the binary strings with n bits. Assume A [O. n - 1I is an array of size n, generate all the strings of length 11 drawn from 0... k - 1.
3. Write algorithm, pseudocode, and code to solve problems like - Give an algorithm for finding maximum element in binary tree.
4. Write algorithm, pseudocode and code to solve AVL Trees problems - Given a height h, give an algorithm for generating the HB(0). HB(0) is generating full binary tree.
5. Prove, is there a min-heap with seven distinct elements so that the pre-order traversal or it gives the elements in sorted order?

6. Travelling Salesperson Problem: Find the shortest path in a graph that visits each vertex at least once, starting and ending at the same vertex'?

List of Laboratory Exercises

1. Finding the length of connected cells of l s (regions) in a matrix of Os and ls.
2. Give an algorithm for finding the maximum clement in binary tree without recursion.
3. Give an algorithm for searching a clement in binary tree.
4. Give an algorithm for finding the diameter of the binary tree. The diameter of a tree (sometimes called the width) is the number of nodes on the longest path between two leaves in the tree.
5. Implement the Algorithm for Building Expression Tree from Postfix Expression
6. Write and implement an algorithm for deleting an arbitrary clement from min heap.
7. Write and implement an algorithm for checking whether a given graph G has simple path from source s to destination d. Assume the graph G is represented using the adjacent matrix.
8. Perform DFS on given graph C.
9. Count the number of connected components of Graph G which is represented in the adjacent matrix.
10. Merging K sorted lists: Given K sorted lists with a total of n elements, write an algorithm to produce a sorted list of all n elements.

Project Based Learning - Provisional List of Projects

1. Design and development of Student attendance system using array data structure.
2. Design and development of Car rental system using Singly linked list (SSL) data structure.
3. Design and development of Inventory management system using suitable data structure.
4. Comparative study of student management system using array, queue, and stack.
5. Design phone dictionary using doubly linked list (DLL).
6. Design and implement of dictionary using hierarchical data structure.
7. Design and implement of expression solver using stack.
8. Design and development quizer (quiz conduction application).
9. Design and development of subject recommendation system.
10. Design and development of Sudoku Solver.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Computer Organisation and Design

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	04 Hours/Week	University Examination	60 Marks	Lecture	04
		Internal Assessment	40 Marks		
		Total	100 Marks	Total	04

Course Objective

This course aims at providing comprehensive understanding of the organization and architecture of modern-day computers, emphasizing both fundamental principles and role of performance parameters in driving computer design.

Prerequisite:

Digital Electronics

Course Outcomes: On completion of the course, students will have the ability to:

1. Analyse the design issues in terms of speed, technology, cost, performance
2. Understand the architecture and functionality of central processing unit.
3. Learn design approaches implementing control unit
4. Discuss the concept of memory organization.
5. Describe structure and functions of I/o module and Peripherals.
6. Infer Performance Enhancement of Processor

Unit I Computer arithmetic and performance **08 Hours**

Computer organization and system architecture, Structure and functions, Von Neumann Architecture, IAS machine. Computer Performance Measurement, Aspects & Factors affecting Computer Performance, MIPS & MFLOPS, designing for performance, fixed and floating-point representations, IEEE 754 format. Booths Algorithm for Signed Multiplication, Restoring and Non-Restoring Division Algorithms.

Unit II Central Processing Unit **08 Hours**

Introduction to x86 microprocessor, Architecture, register organization, Segmentation, Instruction execution cycle, addressing modes, and Instruction set. Instruction Formats, Instruction Types, the Instruction Cycle, and Instruction Pipelining, RISC Vs. CISC Architecture

Unit III Control Unit **08 Hours**

Instruction Cycle & Micro Operations, Functional Requirements & Operations of the Control Unit, Block Schematic & Control Signals, Single Bus Processor Organization, Control Signal example with Micro Operations and Register Transfer. Control Unit Design Methods – Hardwired Control and Micro-Programmed Control Microinstructions & Formats, Control Memory, Microinstruction Sequencing, Sequencing Techniques, Address Generation, Microinstruction Execution, Microinstruction Encoding

Unit IV Memory **08 Hours**

Characteristics of Memory Systems, Internal and External Memory Types. Memory Hierarchy, Principle of Locality, Cache Memory – Basics, Performance Metrics & Improvements, Organization and Mapping Techniques, Handling Cache Misses & Writes, Replacement Algorithms, Cache Controllers.

Unit V I/O organization **08 Hours**

Structure and functions of I/o Module, Peripheral devices and their characteristics, Input-output subsystems, I/O device interface, Programmed I/O, Interrupt driven I/O, DMA, Buses-SCSI, USB

Unit VI Performance enhancement of processors **08 Hours**

CPU Performance and its Factors, Evaluating Performance, Enhancing Performance - Pipeline Processing, instruction pipelining, pipeline stages and hazards, The ARM Cortex–A8 and Intel Core i7 Pipelines, Parallel Processing Concepts - Flynn's classifications, Cache coherence in multiprocessor systems, Specialized Architectures - Multi-core systems

Textbooks

1. William Stallings. "Computer organization and architecture: designing for performance". Pearson Education India, 2010
2. Carl Hamacher, Zvonko Vranesic and Safwat Zaky. "Computer Organization", McGraw Hill, 2011.
3. Computer System Architecture M. M. Mano: 3rd ed., Prentice Hall of India, New Delhi, 1993.
4. Computer Architecture and Organization, John P. Hayes.

Reference Books

1. A. S. Tanenbaum "Structured Computer Organization", 4th Edition, Prentice Hall of India, 1991 ISBN: 81–203–1553–7.
2. Computer Organization and Design: The Hardware/Software Interface, David A. Patterson and John L. Hennessy.

List of Assignments

1. Describe structure of IAS computer with neat block diagram.
2. Describe architecture of 8086 with neat block diagram.
3. Draw and Explain Hardware implementation of Booth's algorithm for signed number multiplication
4. Draw flowchart for Booth's Algorithm for multiplication and perform 9 x -6
5. Draw and Explain Hardwired control unit using a. Delay element method or b. Sequence Counter method
6. Explain Direct mapping and set associative mapping of cache memory in detail with suitable example

List of Laboratory Exercises

1. Write an Assembly Language program to display system time on screen.
2. Write an Assembly Language program to add and subtract two 8-bit numbers.
3. Write an Assembly Language program to determine 2's complement of a number
4. Write an assembly language program for BCD addition and subtraction.
5. To Apply Booth's Algorithm for illustrating multiplication of signed numbers.
6. To design and illustrate Restoring Division Algorithm.
7. To design and illustrate Non-Restoring Division Algorithm.
8. To design 4-bit ALU (VLAB)
9. Study of Memory Design (VLAB)
10. To design Direct mapped Cache and associative cache (VLAB)

Project Based Learning - Provisional List of Projects

Write program to generate assembly code from prefix code.

1. Simulate a word multiplier.
2. Simulate a word divider.

3. Suggest a high-speed addition method and logic for 4-bit addition
4. Design and implement an arbitrary precision four function calculator.
5. Simulate modern traffic control system.
6. Suggest and design a minimal cpu architecture for controlling the washing machine.
7. Write/create/research a tool for benchmarking of a hardware (CPU).
8. Implement quick sort using assembly language.
9. Implement binary search using assembly language.
10. Implement matrix multiplication using assembly language
11. Microprocessor based automatic attendance recorder
12. Microprocessor based furnace temperature controller.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Computer Networks

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	03 Hours/Week	University Examination	60 Marks	Lecture	03
Practical:	02 Hours/Week	Internal Assessment	40 Marks		
		Term Work	25 Marks	Practical	01
		Practical	25 Marks		
		Total	150 Marks	Total	04

Course Objective

This Course emphasis on all theoretical concepts and practical aspects of networking. This course enables the students to understand the networking hardware & concepts through using network simulators.

Prerequisite

Computer System Workshop Technology.

Course Outcomes: On completion of the course, students will have the ability to:

1. Demonstrate the knowledge of computer networking.
2. Elucidate detailed structure of physical and data link layers
3. Demonstrate the knowledge of about Network and Transport Layers.
4. Elucidate the details of Session and Presentation Layers
5. Illustrate the functionality of Application layer
6. Recite the fundamentals of wireless network.

Unit I Fundamentals of Networking **06 Hours**

History of network and internet, need of network, Types of networks, Networking hardware, Information transmission, Transmitter, Receiver, Introduction to networking applications and simulators, Introduction to OSI reference model, Introduction to TCP/IP model, Introduction to internet and working of Internet, Introduction to network security, IEEE standards for networking

Unit II Physical and Data Link Layers **06 Hours**

Ethernet, LLC, MAC, Framing, Channel Allocation schemes, Error Control, Flow Control, Error Detection and correction, Physical Layer protocols, Data Link layer protocols

Unit III Network and Transport Layers **06 Hours**

Network Layer: Network Packet structure and formation, routing algorithms, congestion control algorithms, quality of service, IP Addressing, Subnets, configuring network settings, Network problem solving.

Transport Layers: Segmentation, Congestion control, Connection oriented and connection less services, Network and Transport Layer Protocols.

Unit IV Session and Presentation Layers **06 Hours**

Session Layer: Session management, synchronization, Dialog control, Presentation Layer: Encryption-decryption, Compression, File formats, Translation, Session and Presentation Layer protocols, Session and Presentation Layer protocols.

Unit V Application Layer **06 Hours**

DNS, URL, Data Cache and streaming, Web Applications, Web browser working, Cloud services, User interface and User interaction, Mail systems, Support of file formats, Application Layer protocols

Unit VI Network Security **06 Hours**

Firewall, Types of Firewalls, Cryptography, Symmetric Key Algorithm, Public Key Algorithm, Digital Signatures, Public Key Management, Communication Security, Authentication protocols.

Textbooks

1. Data and computer communications, William Stallings, 10th edition, Pearson
2. Computer networking: a top-down approach, James f. Kurose, Keith w. Ross, 6th edition, Pearson.
3. Computer Networks, Tanenbaum, 5th Edition, Pearson

Reference Books

1. Data communication & networking, Forouzan, 5th edition, McGraw-Hill.
2. Computer Networking Beginners Guide, Russell Scott, 1st edition, Stefano Cardinale

List of Assignments

1. Explain in detail the types of Networks.
2. Consider the real time scenario to explain the error correction and detection.
3. Compare: Connection oriented and Connection less services
4. Consider the real time scenario to explain the role of Session and Presentation Layer in networking
5. Explain in detail: How does the web browser work?
6. Explain in detail: The role of Network Security in computing.

List of Laboratory Exercises

1. Introduction to Computer Network and Network Simulators.
   Networking devices, Addresses, Network Security, Internet working, Network Simulators: Cisco Packet Tracer, Netemul, NetSim.
2. Network configuration of PCs and other networking devices
   Configuring Computer and Router via OS UI and commands, Networking commands
3. Network configuration of PCs and other networking devices using network simulators.
   Observing and configuring PCs, Routers, Switch, Hub, and other networking devices using network simulators
4. Establishment of simple LAN network using real time devices and network simulators.
   Establishment of simple LAN network using actual devices like PCs, Switch, Router and through network simulators
5. Communication between two or multiple systems using network simulators.
   Establishment of network where two network systems can communicate with each other. Use of PCs, Switch etc. in network simulator.
6. Broadcasting using network simulators.
   Establishing network to broadcast the information using network simulator. Use of PCs, Switch and Hub in the network simulator.
7. Establishment of different networks and communication between them using real time devices and network simulators.
   Establishment of different networks and communication between using actual devices like PCs, Switch, Router and through network simulators

8. Understanding Protocols of Transport Layer using Network Simulators
   Understanding Transport Layer protocols TCP, UDP using networking simulators
9. Implementation of Client-Server Architecture for Same Network using Network Simulators and Programming
   Implementing some services of Server with protocols like FTP, HTTP, and others. Use of PCs, Servers, Switch in Network Simulator. Socket Programming: Use of Python, C, C++, Java for implementing Client Server architecture.
10. Implementation of Client-Server Architecture for different Networks using Network Simulators.
    Implementing some services of Server with protocols like FTP, HTTP, and others for different networks. Use of PCs, Servers, Switch and Router in Network Simulator.
11. Understanding Web Browser structure and working
    Web browser structure, source code loading, UI generation, Information loading into application layer, file format support and Complete working of web browser.
12. Establishment of wireless network
    Establishment of wireless networking using actual devices and via network simulator. Use of Laptops and Wifi Router.

Project Based Learning - Provisional List of Projects

1. IP based patient monitoring system
2. Configuring Internet Router
3. Configuring Network Switch
4. Home Automation system using Wi-Fi
5. Wireless Weather monitoring system using Raspberry pi.
6. Smart Traffic control system
7. Smart energy meter for homes
8. Analysis of IPv4/IPv6 protocols
9. Web System Security.
10. Personalized Web Search with Location Preferences

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Software Engineering

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	04 Hours/Week	University Examination	60 Marks	Lecture	04
Practical:	02 Hours/Week	Internal Assessment	40 Marks		
		Term Work	25 Marks	Practical	01
		Oral	25 Marks		
		Total	150 Marks	Total	05

Course Objective:

The course emphasises on the building blocks, importance and need of software engineering. It focuses on the various processes, methods, and practices for developing software which makes software engineering as a specific discipline. This course includes elaboration on each phase of software development life cycle methodologies and practices. It further covers the essential knowledge required to ensure the quality and maintenance of developing and developed software.

Prerequisite:

-

Course Outcomes: On completion of the course, students will have the ability to:

1. Apply the Software Engineering approach to software design and development.
2. Apply the Essential processes of SDLC.
3. Demonstrate the knowledge of requirement elicitation by classifying and documenting the requirements
4. Demonstrate software design by modelling artifacts for gathered requirements & analysis.
5. Apply testing strategies and create test cases and test suites
6. Use the project management concepts and tools for managing software project.

Unit I Overview of Software Engineering & Its Methodologies **08 Hours**

Defining Software Engineering, Software Engineering Principles, Software Engineering Ethics, Software Process, Project, Product and People.
Overview of Software development lifecycle methodologies: Waterfall, Agile, Lean, Iterative, Spiral, DevOps.

Unit II Requirements and Design **08 Hours**

Expressing Requirements, Types of Requirements, Feasibility Study, Elicitation Techniques. Requirements Analysis - Structured Analysis, Object Oriented Modelling, Other Approaches. Requirement Specification, Requirement Validation, Requirement Engineering Tools (CASE).
Software Design: Principles of Software Design, Data Design, Architectural Design, Component Level Design, Object-oriented design, Design Notations, User Interface Design

Unit III Coding and Testing **08 Hours**

Coding Guidelines, Coding Methodology, Programming Practice - Top-down, bottom-up, structured programming, information hiding programming, Code verification Techniques, Introduction to No-Code Development approach and tools.
Testing: Software test Characteristics, Test plan, Test Case Design, Testing Strategies, Testing Techniques, Debugging Process, and strategies.

Unit IV Software Quality and Maintenance **08 Hours**

Quality Concepts, ISO 9126 Quality Factors, Mc Call's Quality Factors, SQA plan, SQA Activities, Software Reviews, Sig Sigma & ISO 9000 Quality Standards, capability maturity model, Software Reliability.

Maintenance: Factors affecting software Maintenance, Types of software Maintenance, Software Maintenance Lifecycle.

Unit V SCM and Re-engineering **08 Hours**

Software Configuration Management - Basics, SCM Planning, Project Library, SCM Process - Configuration Identification, Change Control, Version Control, SCM Tools (CASE).

Software re-engineering: Objectives, Principles of Re-engineering, Levels of Abstraction, Software Re-engineering process Model, Business Process Re-Engineering.

Unit VI Software Planning and Cost Estimation **08 Hours**

Responsibility of Software Project Manager, Project Planning, Project Scheduling, People capability maturity model, Risk Management.

Cost Estimation - Basics, Estimation of Resources, Product Cost Factors, Cost Estimation Process, Constructive Cost Model, Function Point Analysis, Decomposition techniques- Problem based Estimation, Process based estimation, use case-based estimation.

Textbooks

1. Fundamentals Of Software Engineering, Rajib Mall Phi Learning, 02-Apr-2014, Isbn 8120348982, 9788120348981
2. "Software Engineering: Principles and Practices, 2nd Edition by Rohit Khurana, Khurana Rohit · 2010, Vikas Publishing House Pvt Limited", ISBN: 9788125939467
3. Software Engineering principles and practices, Rajesh Narang, 2015 McGraw Hill Education, ASINB014ULF4R8

Reference Books

1. Software Engineering: A Practitioner's Approach, By Roger Pressman and Bruce Maxim, McGraw Hill, 9th Edition, ISBN10: 1259872971
2. Software Engineering, by Ian Sommerville, Pearson; 10th edition, ISBN-10: 0133943038

List of Assignments

1. A mini project will be given to the students based on which they need to prepare the following
   a. Choosing the appropriate SDLC method to develop the given project.
   b. Develop the project plan along with feasibility study and estimations
   c. Prepare the Software requirement Specification document
   d. Prepare the Software Detailed Design Document
   e. Prepare test cases
   f. Use CASE tools to perform all the above tasks.

List of Laboratory Exercises

1. Present a Case study on Agile methodology
2. Present a Case study on DevOps
3. For the given project, perform requirement elicitation using tools and prepare SRS.
4. Prepare case study on Tools used to create Unified Modelling Language.
5. Design Structure Model for the given SRS using UML tool.
6. Design Behaviour model for the given SRS using UML tool

7. Prepare the Testcases using the Junit.
8. Prepare case study on Tools used to prepare project Plan.
9. Prepare a case study on Automated testing Tools.
10. Prepare a case study on Total quality management.

Project Based Learning - Provisional List of Projects

1. Flight Vehicle and Aircraft Systems Engineering.
2. Skyscraper
3. Software piracy protection system
4. e-Learning platform
5. Bug tracker
6. Railway tracking and arrival time prediction system
7. Employee management system
8. Camera motion sensor system
9. Operating System task monitoring application
10. Data leakage detection system

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Programming Technologies and Tools Laboratory – III

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture	01 Hours/Week	Term Work	25 Marks	Lecture	01
Practical	02 Hours/Week	Practical	25 Marks	Practical	01
		Total	50 Marks	Total	02

Course Objective:
The course aims to make students aware of python programming.

Prerequisite:
-

Course Outcomes: On completion of the course, students will have the ability to:
1. Demonstrate the knowledge of using data structures in python.
2. Demonstrate the characteristics of object-oriented Python
3. Perform basic operations on file.
4. Understand and implement error and exception handling
5. Design basic GUI using Python Tkinter
6. Implement database connectivity using MySql and SQLite.

Unit I Python Language Basics **08 Hours**
Python Interpreter, Running IPython Shell, Running Jupyter Notebook, Tab Completion, Introspection, The %run Command, Executing Code from the Clipboard, Terminal Keyboard Shortcuts, About Magic Commands, Language Semantics, Scalar types, Control flow.
Data Structures and Sequences – Tuple, List, built in sequence functions, dict, set, strings.

Unit II Functions, Modules, Packages **08 Hours**
Functions – def statement, returning values, parameters, arguments, local variables, global variables and global statement, Doc Strings, Decorators, lambda, iterators and generators, Modules, Packages.

Unit III Object-Oriented Approach **08 Hours**
Classes - A simple class, defining methods, the constructor, Member variables, calling methods, Adding inheritance, Class variables, Class methods and static methods, Properties, Interfaces, New style classes, Doc strings for classes, Private members.
Scope & Namespaces, object, instantiations, Inheritance, Multiple inheritance, Constructors, operator overloading.

Unit IV File Handling and Error **08 Hours**
Python file handling: File handling modes, Text & Binary Files – Reading, Writing and Delete.
Error and exception handling: Exceptions, Handling Exceptions, Raising Exceptions, Exception Chaining, User-defined Exceptions, Defining Clean-up Actions, Predefined Clean-up Actions.

Unit V Python Tkinter **08 Hours**
Event Driven Programming, GUI frameworks-Tkinter, Windows and windows attribute, Component, Tk widgets-Ttk and Tix widgets, Geometry Management, Events & Binding Functions, simple GUI applications development.

PyGtk – Simple message box, text input dialog box, file selection dialog box.
EasyGUI - Example

Unit VI Backend and Database **08 Hours**

Tornado for windows, building python HTTP web server (GET method), Parameter and Python API (resource and query), Building a JSON GET & POST API, the GET and POST API from JavaScript/HTML
SQLite, MySQL -Environment Setup, Database Connection, CRUD operations.

Textbooks

1. Python 3 Object-oriented Programming Second Edition, Dusty Phillips, Packt Publishing
2. MySQL for Python: Database Access Made Easy,
3. Python GUI Programming with Tkinter, Alan D. Moore, O'Reilly Media, Inc.

Reference Books

1. Introduction to Computation and Programming Using Python, John V Guttag, Prentice Hall of India
2. Python Essential Reference 4th Edition, David Beazley, Pearson Education.

List of Laboratory Exercises

1. Study about Anaconda python software.
2. Write a program to understand the control structures of python
3. Write a program to learn different types of structures (list, dictionary, tuples) in python
4. Write a program to learn concept of functions scoping, recursion, and list mutability.
5. Write a program to understand working of exception handling and assertions.
6. Write a program to perform basic operations on text files.
7. Write a program to implement HTTP server using Python
8. Write a program to implement basic GUI application with database connectivity using SQLite
9. Write a program to learn GUI programming using Tkinter
10. Write a program to implement basic GUI application with database connectivity using MySql

Project Based Learning - Provisional List of Projects

1. Design and development of Mad Libs generator.
2. Design and development of electronic mail system (Read, write, send and delete operations).
3. Design and development of store billing system.
4. Design and development of typing speed check web application.
5. Design and development of windows application for music player.
6. Design and development of windows Quiz Application.
7. Design and development of web application for daily expense tracker.
8. Design and development of student portfolio management & CV generator system.
9. Design and development of windows based to do list or sticky notes.
10. Design and development of assignment plagiarism checker

B.TECH (Computer Engineering)

SEMESTER – IV

COURSE SYLLABUS

Probability and Statistics

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		**Marks**		**Credits**
Lecture:	**03 Hours/Week**	**University Examination**	**60 Marks**	**Lecture**	**03**
		Internal Assessment	**40 Marks**		
		Total	**100 Marks**	**Total**	**03**

Course Objectives:
To equip students with the knowledge of:
- Probability theory and expected value.
- Probability distribution and its applications.
- Multiple regression and ANOVA.

Prerequisite:
Elementary Mathematics

Course Outcomes: On completion of the course, students will have the ability to:
1. Apply Bayes theorem to find probability.
2. Compute mathematical expectations.
3. Identify various theoretical distributions.
4. Use correlation coefficient to interpret numerical data.
5. Use regression to estimate the dependent variable.
6. Apply concept of graph in optimization.

Unit I Probability Theory **06 Hours**
Definition of probability: classical, empirical, and axiomatic approach of probability, Addition theorem of probability, Multiplication theorem of probability, Bayes' theorem of inverse probability, Properties of probabilities

Unit II Random Variable and Mathematical Expectation. **06 Hours**
Definition of random variables, Probability distributions, Probability mass function, Probability density function, Mathematical expectation, Join and marginal probability distributions, Properties of expectation and variance with proofs, Examples

Unit III Theoretical Probability Distributions **06 Hours**
Binomial distribution, Poisson distribution, Normal distribution, Fitting of binomial distributions, Properties of binomial, Poisson and normal distributions, Relation between binomial and normal distributions, Relation between Poisson and normal distributions, Importance of normal distribution

Unit IV Correlation **06 Hours**
Introduction, Types of correlation, Correlation and causation, Methods of studying correlation, Karl Pearson's correlation coefficient, Spearman's rank correlation, Coefficient, Properties of Karl Pearson's correlation coefficient, Properties of Spearman's rank correlation coefficient, Probable errors, Examples

Unit V Linear Regression Analysis **06 Hours**
Introduction, Linear and non-linear regression, Lines of regression, Derivation of regression lines of y on x and x on y, Angle between the regression lines, Coefficients of regression, Theorems on regression coefficient, Properties of regression coefficient

Unit VI Multiple Regression and AVOVA **06 Hours**

Multiple regression & multiple correlation, Analysis of variance (one way, two ways with as well as without interaction)

Textbooks

1. S. C. Gupta, "Fundamentals of Statistics", 46th Edition, Himalaya Publishing House.
2. G. V. Kumbhojkar, "Probability and Random Processes", 14th Edition, C. Jamnadas and co.
3. Murray Spiegel, John Schiller, R. ALU Srinivasan, Probability and Statistics, Schaum's Outlines
4. Kishor S. Trivedi, "Probability, Statistics with Reliability, Queuing and Computer Science Applications", 2nd Edition, Wiley India Pvt. Ltd.

Reference Books

1. Vijay K. Rohatgi, A. K. Md. Ehsanes Saleh, An Introduction to Probability and Statistics, 3 rd Edition, Wiley Publication
2. I.R. Miller, J.E. Freund, and R. Johnson.Fun "Probability and Statistics for Engineers" (4th Edition)

List of Assignments

Six assignments to be given by the course coordinator (Theory)-one from each unit.

Project Based Learning - Provisional List of Projects

Students are expected prepare report on any one topic, write its definition, applications and analyse the hypothetical data. Also, write pseudo code for it, wherever applicable.

1. Bayes theorem
2. Additive and multiplicative law of probability
3. Mathematical expectation
4. Joint and marginal probability distribution
5. Theoretical probability distribution
6. Coefficient of correlation
7. Regression estimates
8. Simple regression model
9. Multiple regression model
10. One way ANOVA
11. Two-way ANOVA
12. Correlation
13. Multiple correlation

Note: - *Students in a group of 3 to 4 shall complete any one project from the above list)

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Models of Computation

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	03 Hours/Week	University Examination:	60 Marks	Lecture	03
Tutorial:	01 Hours/Week	Internal Assessment:	40 Marks	Tutorial	01
		Total	100 Marks	Total	04

Course Objective:
This course enables students to understand any problem by developing abstract models of computing machines and reasoning about their compute efficiency.

Prerequisite:
Discrete Mathematics, Data Structure and Algorithmic Thinking

Course Outcomes: On completion of the course, students will have the ability to:
1. Define and describe formal models of computation
2. Illustrate examples of languages and computational problems appropriate to different models of computation.
3. Demonstrate the relationships between language classes and regular expression.
4. Design grammars and recognizers for different formal languages
5. Design Language Acceptability by Turing Machine
6. Use models of computation to understand the compiler basics

Unit I Basic Concepts and Automata Theory **06 Hours**
Introduction to Theory of Computation- Automata, Alphabet, Symbol, String, Formal Languages, Deterministic Finite Automaton (DFA)- Definition, Representation, Acceptability of a String and Language, Non-Deterministic Finite Automaton (NFA), Equivalence of DFA and NFA, NFA with ε-Transition, Equivalence of NFA's with and without ε-Transition, Finite Automata with output- Moore machine, Mealy Machine, Equivalence of Moore and Mealy Machine, Minimization of Finite Automata

Unit II Regular Languages **06 Hours**
Definition and Examples. Conversion of RE to FA, FA to RE, algebraic laws, applications of RE. Pumping lemma for regular languages and applications. Closure properties of regular Languages Union, Concatenation, Complement, Intersection and Kleene closure. Decidability- Decision properties

Unit III Context Free Grammar (CFG) **06 Hours**
Definition, Derivations, Languages, Derivation Trees and Ambiguity, Regular Grammars-Right Linear and Left Linear grammars, Conversion of FA into CFG and Regular grammar into FA, Simplification of CFG, Normal Forms- Chomsky Normal Form (CNF), Greibach Normal Form (GNF), Chomsky Hierarchy, Programming problems based on the properties of CFGs

Unit IV Push Down Automata (PDA) **06 Hours**
Introduction, Pushdown Automata (PDA), Transition Diagrams, Functions and Tables, Deterministic Push-down Automata (DPDA) - definition, Nondeterministic Pushdown Automata (NPDA), Equivalence of context free grammars and PDA, properties of context free languages. Introduction to Post Machines (PMs).

Unit V Turing Machine Model **06 Hours**

The Turing Machine Model and Definition of TM, Language Acceptability of Turing Machines, Techniques for Turing Machine Construction, Modifications of Turing Machine, Universal Turing machine, Linear Bounded Automata, Church's Thesis, Halting Problem

Unit VI Basics of Compiler **06 Hours**

Introduction to Natural language Processing, Syntax analysis language definition. Primitive recursive functions – Recursive and recursively enumerable languages – Universal Turing machine. Lexical analyser, Text editor, and searching using RE.

Textbooks

1. Theory Computation, Vivek Kulkarni, Oxford higher education
2. Theory of Computer Science (Automata, Language & Computation) K. L. P. Mishra & N. Chandrasekaran, PHI Second Edition
3. Theory of Computer Science, E.V. Krishnamurthy, EWP Publication
4. Introduction to languages and the theory of computation by Jhon C Martin Mc Graw Hill
5. Introduction to Automata Theory, Languages, and Computation (third edition), by John Hopcroft, Rajeev Motwani, Jeffrey Ullman, Addison Wesley, 2007.

Reference Books

1. Introduction to Automata Theory, Hopcroft Ullman, Languages & Computations, Narosa
2. Introduction to Computer Theory, Daniel A. Cohen, Wiley Publication
3. Theory of Computation, Dexter C. Kozen, Springer Science & Business Media, 2006

List of Assignments

1. Study of JFLAP tool for Constructing FA
2. Construct regular expressions defined over the alphabet $\Sigma = \{a, b\}$, which denote the given languages.
3. Translate the following Mealy machine into its equivalent Moore machine.
4. Write a context-free grammar (CFG) which generates the language L denoted by:(a+ b)* bbb(a+ b) *
5. Construct a PDA that accepts the language defined by the following regular grammar.
6. Design a TM to recognize an arbitrary string divisible by 4, from $\Sigma = \{0, 1, 2\}$.

Project Based Learning - Provisional List of Projects

1. Develop a tool to illustrate the algorithm for converting an arbitrary NFA to a DFA.
2. Develop a tool to draw a transition diagram for any given DFA.
3. Approximation algorithms
4. Greedy algorithms.
5. Enumeration of finite automata
6. Enumeration of PDA
7. Enumeration of Turing machines
8. Ambiguous grammars
9. Disambiguation of Grammars
10. Enumeration of Context-free languages
11. Enumeration of Turing machines
12. Universal Turing machines.
13. Randomized Turing machines
14. NP Complete Algorithm

15. Problem solvability using Reduction
16. Design of TM to emulate a finite automaton
17. Design of TM to emulate a PDA
18. Complexity analysis of encryption algorithms using TM.
19. Design of TM to perform sorting
20. Design TM to perform searching.

Syllabus for Unit Tests

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Computer Operating System

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	04 Hours/Week	University Examination:	60 Marks		
Practical:	02 Hours/Week	Internal Assessment:	40 Marks	Lecture	04
		Term Work	25 Marks	Practical	01
		Practical	25 Marks		
		Total	150 Marks	Total	05

Course Objective;
The course focuses on the concepts of operating systems enabling students to understand and apply the principles, structure and functioning of Operating system.

Prerequisite:
Computer architecture and Design, Data structures and algorithms, Programming Skills.

Course Outcomes: On completion of the course, students will have the ability to:
1. To learn and apply the Concepts of operating system
2. Infer the concept of process, thread and Inter process communication
3. Outline the concept of concurrency and deadlocks.
4. Analyse of Memory Management and Virtual Memory
5. Utilize the concepts of I/O System for communication
6. Illustrate the Issues in real time operating system.

Unit I Functionalities & Services of an Operating System **08 Hours**
Architecture of OS, Goals & Structures of O.S, Basic functions, System Calls & Types, Process Concept, Process Control Block, Linux System calls for Process creation, Inter Process Communication using Shared memory / Message passing.

Unit II Concurrency, Multithreaded programming **08 Hours**
Benefits, challenges, models, Pthreads library in Linux: thread creation, cancellation, thread specific data, Thread pools, Signal handling, Scheduling: Pre-emptive, non-pre-emptive algorithms FCFS, SJF, SRT, RR, Thread scheduling: contention scope, Pthread support for scheduling

Unit III Deadlock **08 Hours**
Principles of deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, System calls like signal, Producer Consumer problem (multi-threaded) example Deadlock characterization, Resource graph, Avoidance & Prevention, Safe state, Banker's algorithm, recovery schemes

Unit IV Storage Management **08 Hours**
Memory management, logical v/s physical address space, Segmentation, Paging, Page table structures, Virtual memory, Page replacement strategies, File Systems, file operations, types, access methods, Directory structure, Mounting file systems.

Unit V I/O Systems **08 Hours**
File concept, Access methods, Directory structure, Filesystem mounting, Protection, Directory implementation, Allocation methods, Free-space management, Disk scheduling, Disk management, Swap-space management, Protection.

Unit VI Issues, Protection and Security **08 Hours**

Features of real-time kernels. Real-time CPU scheduling and real-time performance issues, Goals of protection; Domain of protection; Protection models; Security, problems, and threats; Authentication; and Encryption.

Textbooks

1. Operating System Concepts, 9th edition Peter B. Galvin, Greg Gagne, Abraham Silberschatz, John Wiley & Sons, Inc.

Reference Books

1. Modern Operating Systems -By Andrew S. Tanenbaum (PHI)
2. Operating Systems 5th Edition, William Stallings, Pearson Education India

List of Assignments

1. Write in detail about the Quality (Computer Architecture) based on Features and Functionality of latest OS.
2. Discuss in detail the Concurrency mechanism and Multithreaded programming achieved in latest any OS.
3. Explain the mechanism of process and processor management in Unix/Linux OS
4. Discuss in detail the mechanism used for memory management in Linux OS
5. Elaborate in detail the user interface concepts of Linux OS.
6. Write in detail about the Quality (Security Threats) based on Features and functionality of latest OS

List of Laboratory Exercises

1. Basic functionalities and functions of operating system.
2. Write Shell Script to copy the file system from two directories to a new directory in such a way that only the latest file is copied in case there are common files in both the directories.
3. Implementation of FCFS (First Come First Serve) CPU Scheduling.
4. Implementation of SJF (Shortest Job First) CPU Scheduling.
5. Implementation of Round Robin (RR) CPU Scheduling.
6. Producer Consumer Problem Using Semaphores
7. Bankers Algorithm for Deadlock Avoidance
8. Algorithm for Deadlock Detection
9. Page Replacement Algorithms FIFO and LRU
10. Implement Virtualization strategy related to resources.

Project Based Learning - Provisional List of Projects

1. Explore the architectures, features, and functions of open-source operating systems
2. Design the Processes and thread management with deadlock's, synchronization
3. Design Pre-emptive Priority Scheduling algorithm implementation in any language.
4. Java program to analyse page fault for a given page frame using NRU with paging.
5. The project on simulating the multiprogramming of a specific operating system and dealing with CPU scheduling and Job scheduling.
6. Design the project that computes FCFS, SSTF, and SCAN disk-scheduling algorithms
7. Operating Systems mini project to explore the different algorithms of main memory page replacement
8. Develop a client server application to show the inter process communication.
9. Build a file system using the FUSE library.
10. Write a shell interpreter for LINUX.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Database Management System

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	04 Hours/Week	University Examination:	60 Marks		
Practical:	02 Hours/Week	Internal Assessment:	40 Marks	Lecture	04
		Term Work	25 Marks	Practical	01
		Practical	25 Marks		
		Total	150 Marks	Total	05

Course Objective:

Introduction to databases mainly focus on relational models and relation database design. The course enables students with the knowledge models, design paradigms and structured query language. This course introduces students to Semantic Modelling, principles of database management systems (DBMS), DBMS architecture, Database Design, data storage and query processing and transaction management. Further, the course also introduces advanced database systems.

Prerequisite:

Mathematics for Computing-I, Data Structure and Algorithmic Thinking.

Course Outcomes: On completion of the course, students will have the ability to:

1. Design a Relational Database by applying the principles of Database Design
2. Compare and Contrast File Processing and Database Processing
3. Convert the Database Design into Relational Tables as per the application requirement
4. Apply the normalisation to Database Design.
5. Use and relate the concept of transaction, concurrency control and recovery in database.
6. Write queries and commands using Structured query language (SQL)

Unit I Overview of Database Systems: **08 Hours**

Knowing Database and Database Management system (DBMS), Role and Advantages of DBMS, Problems with Traditional File System Processing, types of databases, Database System Architecture - Conceptual Level, External Level, Internal Level, Mappings, Database Users, 2 Tier Architecture and 3 Tier Architecture. Introduction to Relational Databases: Relations and Relvars, Defining Relations, Optimisation.

Unit II Relational Model: **08 Hours**

Values Vs Variables, Types Vs Representations, Type Definition, Operators, Type Generators, Tuples, relation types, Relation Values, Relation variables, Relational Algebra - Syntax and Semantics, Operations, Relational Calculus - Tuple Calculus, Calculus Vs Algebra, Integrity - Predicates and Propositions, Checking Constraints, Constraint classification scheme, Views - Definition, View retrievals and Updates

Unit III Database Design: **08 Hours**

Semantic Modelling - Entity Relationship and Extended Entity Relationship model, Functional Dependencies - Trivial and non-trivial dependencies, Closure of a set of dependencies, closure of a set of attributes, Boyce Codd Normal Form, Normalisation - 1NF, 2NF, 3NF, BCNF, higher Normal Forms.

Unit IV Storage and Querying: **08 Hours**

JBOD, RAID, Files, Data Dictionary storage, Storage Access, Indexing & Hashing - Basics, Ordered Indices, B+ Tree index Files, B Tree Index Files, Multiple Key Access, Static Hashing and Dynamic Hashing, Bitmap Indices.

Querying - Measures of Query Cost, Selection Operation, Sorting, Join Operation.

Unit V Transaction Management: **08 Hours**
Transactions Overview, Transaction Properties, Transaction Log, Concurrency control - Lost Updates, Uncommitted Data, the scheduler, Locking Methods, Time Stamping Methods, Recovery, Isolation Levels, System Recovery, Media Recovery, Savepoints, Serializability.

Unit VI Overview of Advanced Databases: **08 Hours**
Object Oriented Database, Distributed Databases, Logic Based Databases, Temporal Databases, Decision support systems. The Information System Design: System Development Life Cycle, Database Lifecycle, Conceptual Design, Logical Design, Database Design Strategies, Centralized vs Decentralised Design, NoSQL

Textbooks

1. Database System Concepts by Abraham Silberschatz, Henry F. Korth, and S. Sudarshan, 6th Edition, McGraw-Hill Education, 2010.

Reference Books

1. C.J. Date, An Introduction to Database Systems, 8/e, Pearson Education, 2004.
2. Peter Rob and Carlos Coronel, Database System- Design, Implementation and Management (7/e), Cengage Learning, 2007.

List of Assignments
The assignments will be based on a mini project for developing a Database for a system like Student Database Management system, Online Retail Database, Medical record Database etc. There will be tasks that will be allocated to the students like

1. Use the Relational Algebra.
2. Design the EER model
3. Convert the EER model to Relational Tables
4. Apply the Normalisation
5. Create the Tables using SQL and using graphical database building tools.
6. Populate and retrieve the data from tables based on requirements.

List of Laboratory Exercises

1. To study about the Database Management Software Tools and Structured Query Language
2. To Write Data Definition Language queries.
3. To Write Data Query Language queries.
4. To Write Data Control Language queries.
5. To query relational tables using nested queries and Aggregate functions.
6. To perform queries using views.
7. To demonstrate the different types of Joins.
8. Find Results for the questions asked related to the given relational Schema: Employee and Department.
9. Find Results for the questions asked related to the given relational Schema: Publications.
10. To Use MongoDB and perform CRUD operations on it.

Project Based Learning - Provisional List of Projects
RDBMS Design and implementation of various Management database systems:

1. Medical Health record management system

2. Patient detail management system
3. Student Management System
4. On-Demand Online Video Streaming
5. Sports
6. Finances Management System
7. Grocery Management System
8. Weather Management System
9. Web Database system
10. E-commerce Database system.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Wireless Communication

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	04 Hours/Week	University Examination:	60 Marks	Lecture	04
Practical:	02 Hours/Week	Internal Assessment:	40 Marks		
		Term Work	25 Marks	Practical	01
		Oral	25 Marks		
		Total	150 Marks	Total	05

Course Objective:
The course focuses on the fundamentals of wireless communications and provides an overview of existing and emerging wireless communication Technology and networks.

Prerequisite:
Computer Network, Physics for Computing Systems

Course Outcomes: On completion of the course, students will have the ability to:
1. Use of basic concepts and principles of wireless communication
2. Illustrate knowledge of Medium Access Control
3. Recite the working of Telecommunication Systems
4. Recite the functioning and use the satellite and broadcast systems
5. Apply and use Wireless networks.
6. Illustrate the working of Mobile IP and TCP in wireless communication.

Unit I Introduction to Wireless Communication **08 Hours**
Wireless Communication Applications, Open research topics, reference model, Frequencies for Radio transmission – regulations, Signals, Antennas, Signal Propagation – Path Loss of radio signals, Signal propagation effects, multi-path propagation, multiplexing – Space Division Multiplexing, Frequency division multiplexing, time division multiplexing, code division multiplexing.
Modulation – Amplitude shift keying, Frequency shift keying, Phase shift keying, Multi carrier modulation.
Spread Spectrum – Direct sequence spread spectrum; Frequency hopping spread spectrum.

Unit II Medium Access Control **08 Hours**
Specialized MAC requirement, Hidden terminal, and exposed terminals, near and far terminals, SDMA, FDMA, TDMA – Fixed TDM, Classical ALOHA, Slotted ALOHA, Carrier Sense multiple access, Demand assigned multiple access, packet reservation multiple access, Reservation TDMA, Multiple access with collision avoidance, Polling, CDMA – Spread Aloha multiple access, Comparison of SDMA/TDMA/FDMA/CDMA

Unit III Telecommunication Systems **08 Hours**
GSM – Mobile services, System architecture, Radio Interface, Protocols, Localisation & calling, Handover, Security.
DECT – System architecture, Protocol architecture
UMTS and IMT – 2000 – UMTS system architecture, UMTS radio interface

Unit IV Satellite Systems and Broadcasting Systems **08 Hours**
Primer – Orbital aspects, GEO, LEO, MEO, Line of Sight, Routing, Localisation, Handover, Examples.
Broadcast Systems – Cyclical repletion of data, Digital Audio broadcasting – Multimedia object transfer protocol, Digital Video broadcasting – data broadcasting, high speed internet access.

Unit V Wireless LAN **08 Hours**

Infra-red Vs Radio Transmission, Infrastructure and Ad-hoc network, IEEE 802.11 – System Architecture, Protocol Architecture, Physical layer, Medium Access control layer, MAC management, 802.11a, 802.11b.

Bluetooth – Architecture, Radio Layer, Baseband Layer, Link Manager protocol, L2CAP, Security, IEEE 802.15.

Unit VI Mobile Communication Layers **08 Hours**

Mobile IP – Entities and terminologies, IP packet delivery, Agent discovery, Registration, Tunnelling, and encapsulation, IPV6, DHCP.

Traditional TCP – Congestion control, slow start, fast retransmit/fast recovery, Indirect TCP, Snooping TCP, Mobile TCP.

Introduction to 4G, LTE network and 5G communication.

Textbooks

1. Mobile Communications, 2nd Edition, Jochen H. Schiller, Pearson Education, 2003

Reference Books

1. Wireless Communication, Theodore S. Rappaport, Prentice Hall
2. Andreas.F. Molisch, ―Wireless Communications‖, John Wiley – India, 2006.
3. Wireless Communications and Networking, Vijay Garg, Elsevier
4. Wireless Communication –Andrea Goldsmith, Cambridge University Press, 2011
5. David Tse and Pramod Viswanath, ―Fundamentals of Wireless Communication, Cambridge University Press, 2005
6. Dennis Roddy, Satellite Communications Systems, John Wiley & Sons, Ltd 5th Edition

List of Assignments

1. Discuss Comparison of Wireless Technologies in Industrial Application
2. What medium access methods are in use today in wireless networks.
3. Discuss any one System architecture of any latest telecommunication System.
4. Discuss any one latest Digital Audio broadcasting techniques
5. Elaborate latest Protocol Architecture used in practical world.
6. Introduction to 5G networks in Mobile Communication.

List of Laboratory Exercises

1. Introduction to wireless simulation.
2. Configuring wireless networking devices using simulation tool.
3. Establishing wireless LAN network using simulation tool.
4. Configuring wireless router using simulation tool.
5. Case studies on different wireless generations
6. Case studies on IEEE 802.11 Wireless LAN
7. Case studies on different wireless generations.
8. Case studies on short range wireless network
9. Realistic Studies on Wireless Structural Control.
10. Case Studies of Wireless LAN Problems.
11. Case studies on IEEE 802.11 Wireless LAN
12. Case Study of Security Issue in the Wireless Communication System.

Project Based System - Provisional List of Projects

1. Vehicle Tracking System.
2. Accident Identification System.
3. Wireless Camera Position System.
4. Remote Home Security System.

5. Wireless Voting Machine.
6. Wireless Security System.
7. Video Signal Transmitter.
8. Audio Signal Transmitter.
9. RFID based Ambulance Flashing Light with Beeper
10. Bluetooth based Garage Door Opening

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Programming Technologies and Tools Laboratory - IV

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture	01 Hours/Week	Term Work	25 Marks	Lecture	01
Practical	02 Hours/Week	Practical	25 Marks	Practical	01
		Total	50 Marks	Total	02

Course Objective
The course enables students to implement object-oriented designs with Java.

Prerequisite
Programming Technologies and Tools Laboratory – I, Programming Technologies, and Tools Laboratory – II, Paradigms of Programming.

Course Outcomes: On completion of the course, students will have the ability to:
1. Analyse the basics of Java programming.
2. Identify class, objects for real time problems.
3. Make use of constructor, Garbage Collector, and methods of string class.
4. Explore the concept of inheritance and polymorphism with the help of real time applications.
5. Handle the exception with exception handling mechanism and multi-threading programming.
6. Design the graphical user interface by using Applets and AWT.

Unit I JAVA Primer **08 Hours**
Introduction: History and Features of Java, Internals of Java Program, Difference between JDK, JRE and JVM, Internal Details of JVM.
Basics of Java Language-Variable and Reserve / Keywords present in Java, Primitive Data types, Java Operators, Decision making and branching statements in Java

Unit II Classes, Objects and Methods: **08 Hours**
Creating a Class, Visibility/Access Modifiers, Encapsulation, Methods: Adding a Method to Class, returning a Value, adding a Method That Takes Parameters, 'this' Keyword, Method Overloading, Object Creation, Using Object as a Parameters, Returning Object, Array of Objects, Memory Allocation: 'new', Static Data Members, Static Methods,

Unit III Constructors, Destructors and String Handling **08 Hours**
Use of Constructor, Characteristics of Constructors, Types of Constructors, Constructor Overloading, Constructor with Default Arguments, Symbolic Constants, Garbage Collection, Destructors and Finalizers.
String Handling: String: Immutable String, String Comparison, String Concatenation, Substring, Methods of String class, String Buffer class, StringBuilder class, Creating Immutable class, to String method

Unit IV Inheritance and Polymorphism **08 Hours**
Use of Inheritance, Types of Inheritance in Java, Role of Constructors in inheritance, Polymorphism in OOP, Types of Polymorphism, static and dynamic polymorphism, Overriding Super Class Methods. Use of "super" keyword. Interfaces, Implementing interfaces.

Unit V Exception Handling and Multithreaded programming **08 Hours**

Exception Handling: try and catch block, catch block, Nested try, finally block, throw keyword, Exception Propagation, throws keyword, Exception Handling with Method Overriding, Custom Exception.

Introduction to threads, life cycle of a thread, thread states, thread properties, methods in Threads and Runnable, setting priority of threads, synchronization and inter thread communication Life Cycle of a Thread

Unit VI Designing Graphical User Interfaces in Java **08 Hours**

Applet and its use Design Patterns using Applet and JApplet. Run Applet application by browser and applet tool. Applet Architecture. Parameters to Applet Life Cycle of Components and Containers Basics of Components Using Containers Layout Managers and userdefined layout. BorderLayout, FlowLayout, GridLayout, GridbagLayout, BoxLayout. AWT Components Adding a Menu to Window Extending GUI Features Using SWING Components Designing GUI. Advanced swing components like JProgressbar, JSlider, JRadioButton , JTree, JTable, JToggleButton.

Textbooks

1. E. Balaguruswamy, "Object Oriented Programming Using C++ and Java", Tata McGrawHill
2. Steven Holzner et al. "Java 2 Programming", Black Book, Dreamtech Press, 2009.

Reference Books

1. Java The complete reference, Herbert Schildt, McGraw Hill Education (India) Pvt. Ltd. 9th edition, 2014, ISBN: 978-0-07-180856-9 (E-book)
2. Object-Oriented Design Using Java, Dale Skrien, McGraw-Hill Publishing, 2008, ISBN - 0077423097, 9780077423094
3. Mitsunori Ogihara, "Fundamentals of Java Programming", Springer; 2018, ISBN 978-3-319-89490-4
4. Brahma Dathan Sarnath Ramnath, "Object-Oriented Analysis, Design and Implementation an Integrated Approach", Springer; 2nd ed. 2015, ISSN 1863-7310 ISSN 2197-1781 (electronic) Undergraduate Topics in Computer Science ISBN 978-3-319-24278-1, ISBN 978-3-319-24280-4.
5. T. Budd (2009), An Introduction to Object Oriented Programming, 3rd edition, Pearson Education, India.
6. J. Nino, F. A. Hosch (2002), An Introduction to programming and OO design using Java, John Wiley & sons, New Jersey
7. Y. Daniel Liang (2010), Introduction to Java programming, 7th edition, Pearson education, India

List of Laboratory Exercises

1. Write a Java program that prompts the user for an integer and then prints out all prime numbers up to that integer.
2. Write a Java program that describes a class person. It should have instance variables to record name, age, and salary. Create a person object. Set and display its instance variables.
3. Write a Java program that creates a class circle with instance variables for the centre and the radius. Initialize and display its variables
4. Write a Java program that counts the number of objects created by using static variable.
5. Write a Java program to demonstrate the constructors in java.
6. Write a Java program to demonstrate the constructor overloading.
7. Write a Java program to display the use of this keyword

8. Write a Java program that checks whether a given string is a palindrome or not.
9. Write an application that creates an interface' and implements it
10. Write a program that can count the number of instances created for the class.
11. Write an application that executes two threads. One thread displays ―Every 1000 milliseconds and other displays ―B every 3000 milliseconds. Create the threads by extending the Thread class.
12. Create an abstract class shape. Let rectangle and triangle inherit this shape class. Add necessary functions.
13. Write an application that shows the usage of try, catch, throws and finally.
14. Write an Applet that displays ―Hello World‖ (Background colour-black, text colour-blue and your name in the status window.)
15. Develop a scientific calculator using swings.

Project Based Learning

1. Airline reservation system
2. Course management system
3. Data visualization software
4. Electricity billing system
5. e-Healthcare management system
6. Email client software
7. Library management system
8. Network packet sniffer
9. Online bank management system
10. Online medical management system

B.Tech.(Computer Engineering)

Programme Curriculum (2021 Course)

Semester- V

Algorithm Design and Analysis

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	03 Hours/Week	University Examination:	60 Marks		
Practical:	02 Hours/Week	Internal Assessment:	40 Marks	Lecture	03
		Term Work	25 Marks	Practical	01
		Practical	25 Marks		
		Total	150 Marks	Total	04

Course Objective: To provide an overview of performance analysis of algorithms

Prerequisite:
The students should have the knowledge of Data Structures

Course Outcomes: On completion of the course, students will have the ability to:

7. Design and Analyse time complexity
8. Design and Analyse space complexity
9. Discuss Divide and Conquer Method
10. Design algorithms using greedy Methods
11. Infer Backtracking
12. Outline NP-Hard and NP-Complete Problems

Unit I Introduction: Algorithm Basics Pseudocode Conventions, Types of Algorithms, Performance Analysis: Space Complexity, Time Complexity, Asymptotic Notations, Performance Measurements	**06 Hours**
Unit II Basic Algorithms: Basics of Probability Theory, Primality testing, Its Advantages and Disadvantages. Elementary Data Structures: Stack, Queues, Binary Trees, Binary Search trees, Heaps, Heap sort, Graphs, Basic Traversal and Search Techniques. Analysis for complexity of all algorithms.	**06 Hours**
Unit III Divide and Conquer: General Method, Binary Search, Finding the maximum and minimum, Merge sort, Quick sort, Performance Measurement, Worst case Analysis. Strassen's matrix multiplication.	**06 Hours**
Unit IV The General Method, Knapsack problem, tree vertex splitting, Job sequencing. Minimum Cost Spanning Trees: Prim's Algorithm, Kruskal's Algorithm, Optimal Merge Patterns, Single-Source shortest path. Dynamic Programming: The General Method, Multistage Graph, all pair shortest path, Single Source shortest path, General Weights, Optimal Binary Search Trees, String Editing, 0/1 Knapsack, Traveling Salesman Problem.	**06 Hours**
Unit V The General Method, The 8- Queens Problem, Sum of Subsets, Graph Colouring, and Hamiltonian Cycles. Branch and Bound: Least Cost (LC) Search, the 15-puzzle Control abstraction of LC Search, Bounding, FIFO Branch and Bound, LC Branch and Bound.	**06 Hours**

Unit VI: Complexities: Nondeterministic Algorithms, the classes NPHard and NP-Complete, Cook's Theorem, NP-Hard Graph Problems, NPHard Scheduling Problems, NP-Hard Code Generation Problems. Approximation Problems Linear Programming, Network Flows, and Intractability: LP Modelling, Production planning LP Modelling Bandwidth allocation, Network flow, Reductions, Checking algorithms, P and NP **06 Hours**

Textbooks

1) ElitzHorowith and SartajSahani, S. Rajasekaran, “Fundamentals of Computer Algorithms”, Galgotia Publications.
2) Alfred Aho, John E. Hopcroft, “Design and Analysis of Computer Algorithms”, Pearson Education References:
3) Thomas Cormen, Charles E Leiserson, Ronald Rivest, “Introduction to Algorithms, Tata Mc-Graw Hill Publication, Second Edition.
4) Rod Stephens, “Essential Algorithms: A Practical Approach to Computer Algorithms”, John Wiley and Sons Publications
5) Jon Kleinberg, Eva Tardos, “Algorithm Design”, Pearson Education 4) Robert Sedgewick, Philippe Flajolet, “An Introduction to the Analysis of Algorithms”, AdisonWesley Publication, Second Edition
6) Steven S. Skiena, “The Algorithm Design Manual”, Springer Publication, Second Edition.

Reference Books

3. Alfred Aho, John E. Hopcroft, “Design and Analysis of Computer Algorithms”, Pearson Education

List of Assignments

The list of class assignments is provided by the course coordinator.

List of Laboratory Exercises

The sample list of laboratory exercises is given below. This can be used as a guideline and course coordinator can recommend the list for practical.

1. Calculate the time complexity of various algorithms.
2. Calculate the space complexity of various algorithms.
3. Compute the complexity of Prim’s Algorithm
4. Compote the complexity of Kruskal’s Algorithms
5. Analyze the 8-Queens Problem.
6. Implement Optimal Binary Search Tree.

Project Based Learning

1. Design a Sudoku using Recursion
2. Design a Phonebook
3. Simulate 15 Puzzle Problem
4. Design Tic Tac Toe
5. Travelling Salesman Problem
6. Design a board for simulating N-Queen Problem
7. Implement Multistage Graphs
8. Prime Number Generator
9. Random Number Generator
10. Devise and algorithm for large sparse matrix multiplication

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Computer and Information Security

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	04 Hours/Week	University Examination:	60 Marks		
Practical:	02 Hours/Week	Internal Assessment:	40 Marks	Lecture	04
		Term Work	25 Marks	Practical	01
		Oral	25 Marks		
		Total	150 Marks	Total	05

Course Objective:
i) To Explain basics of cryptography, how it has evolved, and some key encryption techniques.
ii) To learn security policies such as authentication, integrity and confidentiality.

Prerequisite:
Basic knowledge of computer network.

Course Outcomes: On completion of the course, students will have the ability to:
1. Explain the basics of network security.
2. Compare different techniques of cryptography
3. Discuss details of key and certificate management
4. Discuss details about system security
5. Recite Network and Transport Layer security
6. Apply knowledge of network security and cryptography in real life

Unit I :Introduction: Security Basics **08 Hours**
Information Security Concepts, Threats, Vulnerabilities, Security Attacks and types, Goals of Security, Data Integrity, Confidentiality, Anonymity, Message and Entity Authentication, Authorization, Non-repudiation, Privacy and data protection, Intrusion detection system (IDS): Need, Types, Limitations and Challenges.

Unit II: Symmetric and Asymmetric Key Cryptography **08 Hours**
Symmetric and Asymmetric key cryptography: -Plain Text and Cipher Text, Substitution techniques, Transposition Techniques, Encryption and Decryption, Data Encryption standards, Block Cipher Design Principles, Advanced Encryption Standards, RC4/RC5 algorithm, IDEA, RSA, Blowfish, S-Box theory, EIGamal Cryptosystem. Steganography, COMSEC, TRANSEC

Unit III :Key and Certificate Management **08 Hours**
Hash Functions, Digital Signatures, Digital Certificates, Knapsack Algorithm, Certificate & DDOS-Distributed Dos based authentication, Smart Cards, PKIX model, PKI components and Applications, LDAP protocol, Creating Digital certificates using Java, DNS certificates, Key agreement protocols: STS protocol, Diffie-Hellman Key Exchange

Unit IV:IP Security **08 Hours**
Overview, IP Security Architecture, Authentication Header, Encapsulating Security Payload, Combining Security Associations and Key Management. Web Security: Web Security Requirements, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET).

Unit V: Security in Networks **08 Hours**
Threats in networks, Network Security Controls – Architecture, Encryption, Content Integrity, Strong Authentication, Access Controls, Wireless Security, Honeypots,

Traffic flow security, Firewalls – Design and Types of Firewalls, Personal Firewalls, IDS, Email Security – PGP,S/MIME

Unit VI: Recent Trends in Network Security **08 Hours**

Case studies on Wireshark, OpenSSL and Snort, Overview of Blockchain, Biometric authentication, Secure E-Commerce (ex. SET), Smart Cards, Security in Wireless Communication. Recent trends in IOT security, IDS, Cloud Security.

Textbooks

2. Atul Kahate, "Cryptography and Network Security", 2nd Edition, Tata McGrawHill
3. William Stallings, "Cryptography and Network Security", Pearson Education.

Reference Books

1. Bruce Schneier ,"Applied Cryptography-Protocols, Algorithms, and Source Code in C".
2. William R. Cheswick. Steven M. Bellovin, Aviel D. Rubin, Addison-Wesley. "Firewalls and Internet Security, Repelling the Wily Hacker".
3. J.W. Rittiaghouse and William M.Hancok – Elseviers. "Cyber Security Operations Handbook".
4. Menezes, van Oorschot and Vanstone, "Handbook of Applied Cryptography".

List of Laboratory Exercises

11. Introduction to Cryptography based Security Tools.
12. Write a Program in C/Java to implement symmetric encryption.
13. Write a Program in C/Java to implement asymmetric encryption.
14. Introduction to GnuPG encryption system.
15. Implementation of Decryption techniques using secret key in GnuPG.
16. Implementation of various cryptographic algorithms using HashCalc
17. Study of how Firewall works in computing.
18. Study of how Antivirus works according to offline or online mode.

Project Based Learning

1. User authentication system
2. Keylogger
3. Password strength indicator
4. Practical Implementation of DES
5. LSB steganography
6. RSA encryption algorithm
7. Cryptography using Playfair Cipher
8. Network traffic analysis
9. Ceaser cipher, Vigenere cipher, Simple Columnar Cipher
10. Hybrid cryptography
11. Cryptography using hellman key exchange

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

COMPILER ENGINEERING

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	04 Hours/Week	University Examination:	60 Marks		
		Internal Assessment:	40 Marks	Lecture	04
		Total	100 Marks	Total	4

Course Objective:
Understand processing of programming languages by computers

Prerequisite:
Basic syntax and semantics of programming languages, Theory of Automata

Course Outcomes: On completion of the course, students will have the ability to:

1. Explain fundamentals of compiler and identify the relationships among different phases of the compiler.
2. Compare different type of Parser.
3. Describe Syntax Directed Translation
4. Obtain the Intermediate Code
5. Apply Code Optimization Techniques
6. Demonstrate the use of tools and Technologies for Compiler Construction.

Unit I **08 Hours**

The Phases of a Compiler, Translator issues, why to write Compiler, Compilation process in brief, Frontend and Backend Model, Compiler Construction tools, Interpreter and the related issues, Cross Compiler, Incremental Compiler, Boot Strapping, Byte code Compilers.

Lexical Analysis. The role of Lexical Analyzer, Input Buffering, Specification of tokens, recognition of tokens, and language for specifying Lexical Analysis alphabet, Token, Lexical error, Block schematic of Lexical Analyzer, "Automatic Construction of Lexical Analyzer-(LEX), LEX Specification and Features. Pattern Matching Algorithms and their Optimization, use of LEX.

Unit II **08 Hours**

Syntax Analysis Introduction: Role of Parsers, Review of Context Free Grammar for Syntax Analysis. Top down-RD parser, Predictive parsers, LL (k) parsers, bottom up Parsers - Operator Precedence Parsers, Shift-Reduce: SLR, LR (k), LALR etc. using Ambiguous Grammars. Error Detection and Recovery, Automatic Construction of Parsers (YACC), YACC specifications. Canonical LR parsers, handling of ambiguous grammars, Error Reporting in LL (1), Operator Precedence and LR Parsing, Efficient Generation of LALR (1) sets, Optimization of LR parsers, Optimization of transformations, Detection, Reporting, Recovery and Repair of errors in the Compilation Process.

Semantic Analysis Need of Semantic Analysis, Type Checking and type Conversion Overloading of Functions and Operators, Polymorphic Functions, Unification Algorithm.

Unit III **08 Hours**

Syntax Directed Translation: Syntax Directed Translation Schemes, Implementation of Syntax Directed Translators, Intermediate Code, Postfix Notation, Parse Trees and

Syntax Trees, Three-Address code, Quadruples, and Triples, Translation of Assignment Statements, Boolean Expressions, Translation with a Top-Down Parser.

Unit IV **08 Hours**

Runtime Environment Source Language Issues, Storage Organization and Allocation Strategies, static allocation, stack allocation for Block Structured and non-block structured languages, Activation trees , Activation record, variable-length data, procedure parameters, nested procedures, access to non-local names, procedure Call and return, static and dynamic scope, dangling references ,Symbol Table Organization and Management Entries, Storage allocation, Hash tables, Scope, Compilers for parallel machines, Compilers for functional languages.

Unit V **08 Hours**

Code Optimization

Basic blocks and folding, optimization within iterative loops, global optimization through flow graph analysis, Code-Improving Transformations, Machine Dependent Optimization

Introduction, Classification of optimization, Principle sources Of Optimization, optimization of basic blocks, Loops in flow graphs, Optimizing transformations: compile time evaluation, Common sub-expression elimination, variable propagation, code movement, strength reduction, dead code elimination and loop optimization, Local optimization, Global Optimization: Control and data flow analysis, Computing Global Data flow information: meet over paths, Data flow equations, Data flow analysis, Iterative Data Flow Analysis: Available expressions, Live Range Identification.

Unit VI **08 Hours**

Code Generation Introduction: Intermediate languages, Translation of Declarations & Assignments statements. Design issues of a Code generator, Target machine, Runtime storage Management, Basic blocks, and flow graphs. Issues in code generation, Target machine description, next-use information, register allocation and assignment, Dag representation of basic blocks, Peephole optimization, generating code from a DAG, Dynamic programming, Semantic stacks, Attributed Translation, Analysis of syntax, Directed Translation, Evaluation of expressions, control structures, Procedure calls LEX and YACC

Textbooks

1. A V Aho, R. Sethi, .J D Ullman, "Compilers: Principles, Techniques, and Tools", Pearson Education, ISBN 81 - 7758 - 590 – 8

Reference Books

1. K. Cooper, L, Torczon, "Engineering a Compiler", Morgan Kaufinann Publishers, ISBN 81-8147-369-8.
2. K. Louden, "Compiler Construction: Principles and Practice", Thomson Brookes/Cole (ISE), 2003, ISBN 981 - 243 - 694-4
3. J. R. Levine, T. Mason, D. Brown, "Lex & Yacc", O'Reilly, 2000, ISBN 81-7366 -061-X.
4. S. Chattopadhyay, "Compiler Design", Prentice-Hall of India, 2005, ISBN 81-203-2725-X.
5. D. M. Dhamdhere, Compiler Construction—Principles and Practice, (2/e), Macmillan India
6. Andrew Appel, Modern Compiler Implementation in C, Cambridge University press
7. K C. Louden "Compiler Construction—Principles and Practice" India Edition, CENGAGE
8. Bennett J.P., "Introduction to Compiling Techniques", 2/e (TMH).

Project Based Learning Topics:

1 Lexical Analyzer Using Python
2 Recursive Descent Parser

3	Intermediate Code Generator
4	Parenthesis Checker
5	CLI Calculator Using FLEX And BISON
6	Design A Scanner For C Language Using Lexical Analysis
7	Design Turing Machine For Sorting
8	Parser For C Programming Language
9	Text To Binary Convertor
10	Epsilon NFA to DFA

Syllabus for Unit Tests:

Unit Test- 1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Data Warehousing and Mining

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	03 Hours/Week	University Examination:	60 Marks		
Practical:	02 Hours/Week	Internal Assessment:	40 Marks	Lecture	03
		Term Work	25 Marks	Practical	01
		Practical	25 Marks		
		Total	150 Marks	Total	04

Course Objective:

- To understand data warehouse concepts, architecture, business analysis and tools
- To understand data pre-processing and data visualization techniques
- To study algorithms for finding hidden and interesting patterns in data
- To understand and apply various classification and clustering techniques using tools.

Prerequisite:

- Knowledge of Data Base Management System

Course Outcomes:

Upon completion of the course, the students should be able to:

1. Compare different data mining tools.
2. Design a Data warehouse system and perform business analysis with OLAP tools.
3. Apply suitable pre-processing and visualization techniques for data analysis
4. Apply frequent pattern and association rule mining techniques for data analysis
5. Apply appropriate classification and clustering techniques for data analysis.
6. Demonstrate use of Data mining tools.

Unit I: DATA WAREHOUSING, BUSINESS ANALYSIS AND ON-LINE ANALYTICAL PROCESSING (OLAP) **06 Hours**

Basic Concepts - Data Warehousing Components – Building a Data Warehouse – Database Architectures for Parallel Processing – Parallel DBMS Vendors - Multidimensional Data Model – Data Warehouse Schemas for Decision Support, Concept Hierarchies -Characteristics of OLAP Systems – Typical OLAP Operations, OLAP and OLTP.

06 Hours

Unit II: DATA MINING – INTRODUCTION

Introduction to Data Mining Systems – Knowledge Discovery Process – Data Mining Techniques – Issues – applications- Data Objects and attribute types, Statistical description of data, Data Pre-processing – Cleaning, Integration, Reduction, Transformation and discretization, Data Visualization, Data similarity and dissimilarity measures.

Unit III: DATA MINING - FREQUENT PATTERN ANALYSIS **06 Hours**

Mining Frequent Patterns, Associations and Correlations – Mining Methods- Pattern Evaluation Method – Pattern Mining in Multilevel, Multi-Dimensional Space – Constraint Based Frequent Pattern Mining, Classification using Frequent Patterns

Unit IV: CLASSIFICATION AND CLUSTERING **06 Hours**

Decision Tree Induction - Bayesian Classification – Rule Based Classification – Classification by Back Propagation – Support Vector Machines –– Lazy Learners

Clustering Techniques – Cluster Analysis-Partitioning Methods - Hierarchical Methods – Density Based Methods - Grid Based Methods – Evaluation of clustering – Clustering high dimensional data- Clustering with constraints, Outlier analysis-outlier detection methods.

Unit V: Text and Web Mining **06 Hours**

Text mining: Text Data Analysis and Information Retrieval, Dimensionality Reduction for Text, Feature vector, Bag of words, Tf-idf, Text Mining Approaches, Web mining: Introduction, web content mining, web usage mining, web structure mining, web crawlers

Unit VI: WEKA TOOL **06 Hours**

Introduction, Iris plants database, Breast cancer database, Auto imports database - Introduction to WEKA, The Explorer – Getting started, Exploring the explorer, Learning algorithms, Clustering algorithms, Association–rule learners.

List of Assignment

- Creating a simple data warehouse & performing OLAP operations using simple tools
- Extracting data from any Operational database (ETL) and performing pre-processing tasks
- Performing association mining on large data sets and extracting best possible rules / a case study
- Performing classification and evaluating the efficient model / a case study
- A case study on finding efficient Clusters on very large set of documents data
- A case study on Web mining and Text mining using software tools

Project Based Learning Topics:

1. Implement face recognition attendance system.
2. Implement Color detection system.
3. Implement food recommendation system.
4. Implement Deep Learning Model to identify between human and horse.
5. Implement Housing price prediction system.
6. Implement movie recommendation system
7. Implement Iris database predictor.
8. Implement Email Spam Classifier system.
9. Implement 15 Puzzle Problem using Branch and Bound algorithm
10. Implement Hand sign recognition.
11. Implement Dijkstra's algorithm and find out shortest path between cities.

Textbooks:

- Jiawei Han and Micheline Kamber, ―Data Mining Concepts and Techniques‖, Third Edition, Elsevier, 2012.
- Alex Berson and Stephen J. Smith, ―Data Warehousing, Data Mining & OLAP‖, Tata McGraw – Hill Edition, 35th Reprint 2016.

Reference Books:

- K.P. Soman, Shyam Diwakar and V. Ajay, ―Insight into Data Mining Theory and Practice‖, Eastern Economy Edition, Prentice Hall of India, 2006.
- Ian H. Witten and Eibe Frank, ―Data Mining: Practical Machine Learning Tools and Techniques‖, Elsevier, Second Edition
- G. K. Gupta, "Introduction to Data mining with case studies", PHI, second edition

MICROPROCESSOR AND MICROCONTROLLER

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	04 Hours/Week	University Examination:	60 Marks		
Practical:	02 Hours/Week	Internal Assessment:	40 Marks	Lecture	04
				Practical	01
		Total	100 Marks	Total	05

Course Objective:

1. To develop an understanding of the architecture and functions of microprocessors and micro-controllers.
2. To learn machine language programming & interfacing techniques.

Prerequisite:

Digital Electronics, Computer Organization

Course Outcomes: On completion of the course, students will have the ability to:

1. Describe architecture of microprocessor.
2. Use programmer's model of 80386.
3. Demonstrate concepts of segmentation and paging
4. Comprehend hardware and software interaction and integration.
5. Describe architecture of micro controller.
6. Design microcontroller-based systems.

Unit I Introduction to 80386 **08 Hours**

Concepts of architecture of 80386 DX, Registers, Salient features of 80386 DX, Signal definition, addressing modes, Instruction format, Instruction pipelining in 80386.

Unit II Real and protected mode **08 Hours**

Real mode programming model, Memory addressing in real mode, Interrupt handling and exceptions, switching between real and protected mode, protected mode register model.

Unit III I/O interface **08 Hours**

Programmable peripheral devices and interfacing, Interfacing with Programmable Peripheral Interface 8255, Programmable Interrupt Controller 8259, USART 8251, Programmable Interval Timer 8253, DMA Controller 8237, Block diagram, operating modes, and control words.

Unit IV Multicore Architecture **08 Hours**

Intel 64-bit architecture, Pentium processor functional block diagram, memory management, Multicore Architecture, Bus Connections, core to duo and dual core processors, characteristics and design guidelines.

Unit V Introduction to PIC microcontroller **08 Hours**

Comparison of microprocessor and micro controller, Micro controllers and embedded processors, Overview of PIC Family, PIC Architecture, Registers, and memory in PIC, WREG register, File register, Status register Special Function Registers, RISC architecture in PIC, I/O ports

Unit VI Programming and interfacing of PIC **08 Hours**

Counter and timer programming, interrupt programming, serial communication Programming, Interfacing keyboard and Display, Standard CCP modules and programming

Textbooks

1. Intel microprocessor and peripheral handbook (32 bit) 80386 DX
2. Muhammad Ali Mazidi "PIC Microcontroller and embedded systems",2009, Pearson education
3. 80386 Microprocessor Handbook, Chris H. Pappas, William H. Murray

Reference Books

1. D.V. Hall, "Micro Processor and Interfacing ", Tata McGraw-Hill.
2. Intel 64 and IA-32-bit architectures Software Developer's Manual, Volume 3A, Intel, (Digital Content PDF: 253668.pdf)
3. Microcomputer Systems: The 8086/8088 Family: Architecture, Programming and Design, 2^{nd} ed., Liu &Gibson

List of Laboratory Exercises

1. Write an assembly language program for multiplication of two 16-bit numbers.
2. Write an assembly language program to count number of negative and positive numbers from given array.
3. Write an assembly language program to check whether entered string is palindrome or not.
4. Write an assembly language program to arrange given set of numbers in ascending order.
5. Write an assembly language program to arrange given set of numbers in descending order.
6. Write an assembly language program for 8-bit BCD addition and subtraction.
7. Write an assembly language program (for 80386) for the multiplication of two 32-bit numbers.
8. Write an assembly language program to display current time from system.
9. Write an assembly language program to interface 8051 microcontrollers with keyboard
10. Write an assembly language program to interface 8051 microcontrollers with DC motor

Project Based Learning

1. Home Security System using Bluetooth Technology
2. Ultrasonic Distance Meter with Buzzer indication
3. Alcohol detection system using Bluetooth technology
4. Industrial device controlling using Bluetooth
5. Gate Controller with High Speed Alerting System
6. Celsius Scale Thermometer
7. RFID based Attendance System
8. Fire Extinguishing Robot
9. Temperature and Humidity Measurement

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Programming Technologies and Tools Laboratory V

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	01 Hours/Week	University Examination:	60 Marks		
Practical:	02 Hours/Week	Internal Assessment:	40 Marks	Lecture	01
		Term Work	25 Marks	Practical	01
		Practical	25 Marks		
		Total	200 Marks	Total	02

Course Objective:

1. To provide an insight of Internet programming and how to design and implement complete applicationsover the web.
2. To provide hands-on experience of the recent platform, technologies and design Methodologies used indeveloping web Applications.

Prerequisite:

1. Basics concepts of Object-Oriented Programming
2. “Java” programming language (Core and Advanced Java)

Course Outcomes:

1. Develop WEB 2.0 Rich Internet Applications using the hierarchy of objects in HTML and XML.
2. Design dynamic and interactive web pages using HTML and Ajax performing client validation using Java Script.
3. Use ontology and inference engines in semantic web development.
4. Develop full-fledged Enterprise Applications using HTML, PHP and Java script
5. Implement enterprise beans and understand its benefits.
6. Develop Web services using SOAP.

Unit I Introduction to WEB 2.0 Rich Internet Applications: **08 Hours**

Introduction: From Browsers to Rich Clients – browser drawbacks, A solution – rich clients, Rich clients today. HTML 5: Detecting HTML 5 features – Canvas, video, local storage, web workers, offline Applications, geo location, placeholders, input types, doc type, root, headers, articles, dates and times, navigation and footers. JSF, The WEB Model and XML,

08 Hours

Unit II PHP

Introduction to, Basic rules of PHP Program, Evaluation of Php, Basic Syntax, Defining variable and constant, Php Data type, Operator and Expression. Decisions and loop Making Decisions, looping, Function, Recursive function, String Creating and accessing, String Searching & Replacing String, Formatting String, String Related Library function, array, Element Looping with Index based array. Handling Html Form with Php Capturing Form

Unit III AJAX-I: **08 Hours**

Java Script Fundamentals, Objects in Java Script, Dynamic HTML with Java Script, ASP, Basic communication techniques –AJAX with images, Dynamic script loading, AJAX libraries – jQuery, JSON, JSON versus XML, server-side JSON tools.

Unit IV Semantic Web Technologies: **08 Hours**

Introduction to Semantic Web, Web 3.0 and Semantic Web; why Semantic Web; Impact of Semantic Web; Myths about Semantic Web; Ontologies: Introduction to Ontology; Types of Ontologies, Basic OWL; Class, Properties and Constraints; Ontology development methodology; Ontology tools- SPARQL, Search Engine Optimization (SEO). **Applications of Semantic Web:** Software Agents; Semantic Search; Semantic Web Services; Semantics in Social Networking; SOA, ETL; Web crawling, Page Ranking Algorithm.

Unit V Enterprise JAVA (J2EE): **08 Hours**

Defining the Enterprise, Introducing Enterprise Applications, Creating dynamic content with servlets, using Java server pages (JSP), NetBeans, interacting with relational databases using JDBC,MVC Architecture, JMS, Managing transactions relational databases using JDBC,MVC Architecture, JMS, Managing transactions with JTA/JTS, security aspects in system architecture, J2EE design patterns.

Unit VI: Enterprise Beans: **08 Hours**

Enterprise Bean, Benefits of Enterprise Beans, When to Use Enterprise Beans, Types of Enterprise Beans - Session Bean, Message-Driven Bean, Contents of an Enterprise Bean, Packaging Enterprise Beans in EJB JAR Modules, Naming Conventions for Enterprise Beans, The Lifecycles of Enterprise Beans - Stateful Session Bean , Stateless Session Bean, Creating the Enterprise Bean . Coding the Enterprise Bean Class.

List of Assignment

1. Design the static web pages required for a website using HTML 5.

2. Design the static web pages required for a website using HTML 5.
   (a) Create the Bouncing Ball animation using HTML5 Canvas.
   (b) Create different shape & Bar graph

2. Illustrate XML document processing in Java using XPath and XSLT.

3. Design the web pages required for a website using PHP.

4. Illustrate the use of Java Script in performing client-side validation.

5. Describe and Build Dynamic web Pages using DHTML with Java Script.

6. Create dynamic content with servlets, using Java server pages (JSP), Net-Beans, interacting with relational databases using JDBC.

6. Construct small applications using AJAX

7. Design and build small applications using html, Php, JQuery, JSON and AJAX.

8. Outline and Discuss on Applications of Sematic Web Technologies.

8. Report how to use OWL to develop Ontologies.

9. Construct a Web Crawler

10. Outline and Discuss on implementation of Page Ranking Algorithm

11. Build and Illustrate an Enterprise Applications

12. Build and Apply Web Services using SOAP

Textbooks

1. Eric Van Der Danny Ayers et al, “Professional Web 2.0 Programming”, Wrox Publications, 2007

2. Web Technologies: Black Book, Kogent Learning Solutions Inc. Wiley India Pvt. Ltd.2006

3. David Hunter et al, “Beginning XML”, 4th Edition, Wrox/John Wiley, 2007
4. Mark Pilgrim “HTML 5 Up and Running”, O’REILLY| GOOGLE Press, 2010.
5. Nicholas C Zakas et al, “Professional AJAX”, 2nd Edition, Wrox publications,2007.
6. Thomas Erl ,“SOA: Concepts, Technology and Design”, Pearson, 2005
7. Karin K. Breitman, Marco Antonio Casanova, and Walter Truskowski “Semantic Web: Concepts,Technologies and Applications”, Springer International Edition, 2007.
8. Justin Couch and Daniel Steinberg, Hungry Minds Inc, “Java 2 Enterprise Edition Bible” 2002.

Reference Books

1. Lee W. Lacy, OWL: Representing Information Using the Web Ontology Language, Trafford Publishing,2005.
2. Christopher D. Manning, Prabhakar Raghavan & Hinrich Schutze, Introduction to Information Retrieval,Cambridge university press, 2008

B.Tech. (Computer Engineering)

Programme Curriculum (2021 Course)

Semester-VI

Big Data Analytics

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	04 Hours/Week	University Examination:	60 Marks		
Practical:	02 Hours/Week	Internal Assessment:	40 Marks	Lecture	04
		Term Work	25 Marks	Practical	01
		Practical	25 Marks		
		Total	150 Marks	Total	05

Course Objective: This course focuses on the basic concepts of big data, methodologies for analyzing structured and unstructured data using Hadoop & R.

Prerequisite: Knowledge on Database Management System, Distributed Computing, Data Mining, File System, Statistics.

Course Outcomes: On completion of the course, students will have the ability to:

1. To outline the Big Data fundamentals and various data challenges getting introduced.
2. To accurately and objectively examine, and critically investigate Big Data Technologies and Infrastructure concepts, theories.
3. To apply the novel architectures and platforms introduced for Big data, in particular Hadoop and MapReduce.
4. Being able to describe and apply the Data Analytics lifecycle to Big Data projects
5. To analyze methods and algorithms with respect to data and application requirements, and make appropriate design choices when solving real-world problems
6. To apply secured policy oriented techniques for solving real-world problems.

Unit I : Introduction to Big Data: 08 Hours

Types of Data, Characteristics of big data Information, Phases: capture, organize, integrate, analyze, and act, Defining Structured Data, exploring sources of big structured data, Understanding the role of relational databases in big data, Defining Unstructured Data, exploring sources of unstructured data, Understanding the role of a CMS in big data management, Integrating data types into a big data environment., Orientation to SQL on Big Data

Unit II :Statistics & Analytics 08 Hours

Numerical Data, Summary Statistics, Sampling and Confidence Intervals, From Population to Sampled Data, Different Types of Biases, Probability, Statistical Inference, Conditional Probability and Bayes Rule, Random Variables, Linear Regression, Bayesian Modeling, , Text Analytics, Characteristics of big data analysis, Linear Models, Decision trees, Neural Networks, SQL SELECT Essentials

Unit III : Predictive Analysis Process and R 08 Hours

Data Analytics Lifecycle, Discovery, Data Preparation, Model Planning, Model Building, communicate results, Operationalize, Building a Predictive model. Introduction to R, R graphical User Interfaces, Data import and Export, Dirty Data, Data Analysis, Statistical Methods for Evaluation, Linear regression with R, Clustering with R hypotheses testing., Filtering Data

Unit IV : Big Data Storage and Computing Platforms: 08 Hours

Distributed computing for big data, the changing economics of computing, the problem with latency, Demand meets solutions, Physical infrastructure layer, Big Data Stack, Layer 0: Redundant
Physical Infrastructure, Layer 1: Security Infrastructure, Layer 2:
Operational Databases, Layer 3: Organizing Data Services and Tools,Layer 4: Analytical Data Warehouses, virtualization in big data,Traditional RDBMS, NoSQL, Design Goals for MongoDB, Grouping and Aggregating Data

Unit V : Hadoop: 08 Hours

The Core of Hadoop: MapReduce, Hadoop's Lower Levels: HDFS and MapReduce, Improving Programmability: Pig and Hive, Common building blocks, Key/value pairs, Getting your data into Hadoop, Other Hadoop Components, Hadoop in action. Sorting and Limiting Data

Unit VI : Case Studies and Data Privacy: 08 Hours

Defining Privacy and Security, Data and User Anonymization Fraud and Big Data, Risk and Big data, Credit Risk Management, Big Data & Algorithm Trading, Advertising and Big Data, The Privacy Landscape, Rights and Responsibilities, Case Study: Recommendation Engine, Sentiment Analysis and Digital Marketing, Healthcare applications.

Textbooks

John Wiley & Sons, "Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data", EMC Education Services, 2015

Reference Books

1. Vignesh Prajapati, "Big Data Analytics with R and Hadoop," Packt Publishing Ltd, 2013,

2. Big Data Analytics, Introduction to Hadoop, Spark, and Machine-Learning,MC Hill ,Rajkamal 2019

List of Assignments

1. Ecosystems of Big Data and The Hadoop Framework
2. List different scenarios to load data into Hadoop, how to load data at rest, in motion and from sources
3. Import data from a relational database into HDFS
4. Describe Flume and its uses
5. Describe the Map reduce Model
6. Overview of Text Analytics and Streams

List of Laboratory Exercises

1. Demonstrate the concept of variables and data types in R and Create Vectors, Matrices, Lists, Arrays, factors and data frames using it.
2. Import data, copy data from Excel to R, consider .csv, & .txt formats and performs Subset, Cbind, Rbind commands on it

3. Write R program to find the levels of factor of a given vector, also create a factor corresponding to height of women data set, which contains height and weights for a sample of women.
4. Write R program to count the number of NA values in a data frame column. Write R program to find elements come only once that are common to both given data frames.
5. Write R script to make Bar chart (), Pie chart () and Box Plot (titanic data set) on given datasets.
6. Produce stratified Boxplots, Histograms and Scatterplots using different datasets.
7. Demonstrate & calculate Binomial Distribution, Poisson Distribution and Normal Distribution for given values.
8. Write R script to build Linear Regression Model using given dataset.
9. Perform Apriori Analysis using arules package.
10. Case Study: Implement K-Means Algorithm on worlddata dataset and visualize the clusters.

Project Based Learning

1. Twitter Sentiment Analysis on Russia-Ukraine War .
2. Fine-tuning BERT model for Sentiment Analysis.
3. Aspect Modelling in Sentiment Analysis.
4. Sentiment Analysis of Hindi Text.
5. Flipkart Reviews Sentiment Analysis .
6. Implement a word cloud on any e- commerce dataset.
7. With collaborative filtering build recommendation engine.
8. Use Shiny package to implement interactive dashboard.
9. Exploratory data analysis using Shiny package.
10. Predictive data analysis using Shiny package.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Essentials of Internet of Things

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	04 Hours/Week	University Examination:	60 Marks		
Practical:	02 Hours/Week	Internal Assessment:	40 Marks	Lecture	04
		Term Work	25 Marks	Practical	01
		Oral	25 Marks		
		Total	150 Marks	Total	05

Course Objective:

To implement IOT using Python-based IDE (integrated development environments) for the Raspberry Pi.

Prerequisite:

Digital Electronics, Wireless Communication, Microprocessor and Micro controller, Computer Networks, Python programming.

Course Outcomes:

1. Identify the components of IOT.
2. Analyse networking protocols in IOT.
3. Evaluate the connectivity technologies in in IOT.
4. Describe Wireless Sensor networks.
5. Demonstrate applications of IOT.
6. Comprehend Architecture of Raspberry Pie.

Unit I Introduction to IOT **08 Hours**

Technologies involved in IoT development, Infrastructure, Overview of IoT supported Hardware platforms such as: Raspberry pi, ARM Cortex Processors, Arduino and Intel Galileo boards, Types of Sensors and Actuators.

Unit II Networking in IOT **08 Hours**

Internet web and Networking technologies, Functional components of IOT, IOT gateways, IP V4 and IPV6, IOT protocols- MQTT , Components and methods, CoAP, XMPP,AMQP.

Unit III Connectivity Technologies **08 Hours**

Communication protocols like 802.15.4, Zigbee, 6LOWPAN, Bluetooth, Wireless HART, RFID, Z Wave. Remote monitoring and sensing, remote controlling, and performance analysis

Unit IV Sensor Networks **08 Hours**

Wireless sensor networks, Sensor nodes, Sensor web, social sensing in WSN, Target Tracking, WSN coverage, Stationary and mobile WSN, Mobile nodes, Role of M2M in IoT, SCADA (Supervisory control and data acquisition)

Unit V Smart cities and Smart Homes **08 Hours**

Privacy and Trust in IoT-Data-Platforms for Smart Cities, Smart Economy, Data Aggregation for the IoT in Smart Cities, Data fusion, Smart parking, smart home infrastructure, HAN network standards and architecture

Unit VI Introduction to Raspberry Pie **08 Hours**

Specifications, Architecture, Basic set up, Integration of sensors and actuators, Capturing Image with Raspberry Pi, Implementation of IOT with with Raspberry Pi

Textbooks

1. Vijay Madisetti and ArshdeepBahga, “Internet of Things (A Hands-on-Approach)”, 1 st Edition, VPT, 2014
2. Adrian McEwen, Hakim Cassimally, “Designing the Internet of Things”, November 2013, John Wiley and Sons
3. Simon Monk, “Programming the Raspberry Pi: Getting Started with Python”, January 2012, McGraw Hill Professional

Reference Books

1. Francis daCosta, “Rethinking the Internet of Things: A Scalable Approach to Connecting Everything”, 1 st Edition, Apress Publications, 2013
2. CunoPfister, Getting Started with the Internet of Things, O‟Reilly Media, 2011, ISBN: 978-1-4493- 9357

List of Laboratory Exercises

1. Identify different types of sensors and Actuators
2. To Install Raspian on SD card
3. Use Python-based IDE (integrated development environments) for the Raspberry Pi
4. LED Interfacing with Raspberry Pi
5. To interface push button with Raspberry Pi
6. To interface Bluetooth Raspberry Pi
7. To interface temperature sensor with Raspberry Pi
8. To interface camera with Raspberry Pi to capture image
9. Study MQTT protocol
10. Project

Project Based Learning

11. Home Automation System.
12. Car Parking Management System.
13. Health Monitoring System.
14. Air & Noise Pollution Monitoring System.
15. Smart Street Light Monitoring System.
16. Fall detection using IOT
17. Green House Automation using IOT
18. Water level monitoring using IOT
19. Smart security camera
20. Smart alarm clock using Google calendar.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Graphics Techniques and GPU

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	04 Hours/Week	University Examination:	60 Marks		
Practical:	02 Hours/Week	Internal Assessment:	40 Marks	Lecture	04
		Term Work	25 Marks	Practical	02
		Oral	25 Marks		
		Total	150 Marks	Total	05

Course Objective:

1. To familiarize with the universal concepts of computer Graphics Programming.
2. To study how interaction is handled in a graphics system
3. To study how to manipulate graphics object by applying different transformations
4. To provide the programmer's perspective of working of computer graphics

Prerequisite:

Programming language c, C++

Knowledge of data structures and algorithm is preferable.

Course Outcomes: On completion of the course, students will have the ability to:

1. Understand fundamentals of the modern GPU programming pipeline
2. Apply Essential mathematics in computer graphics
3. Describe common data structures and manipulate geometry
4. Implement Basic shading techniques
5. Implement Basic image-processing techniques
6. Understand How the human visual system plays a role in interpretation of graphics

Unit I **08 Hours**

Introduction – historical evolution, issues and challenges, graphics pipeline, hardware and software basics

Object representation – boundary representation, splines, space partitioning

Unit II **08 Hours**

Modeling transformations – matrix representation, homogeneous coordinate system, composition, 3D transformations

Illumination and shading – background, simple lighting model, shading models, intensity representation, color models, texture synthesis

Unit III **08 Hours**

3D viewing – viewing pipeline, view coordinate system, viewing transformation, projection, window-viewport transformation

Clipping and hidden surface removal – clipping in 2D. 3D clipping algorithms, hidden surface removal

Rendering – scan conversion of line, circle, fill-area and characters, anti-aliasing

Unit IV **08 Hours**

Review of Traditional Computer Architecture – Basic five-stage RISC Pipeline, Cache Memory, Register File, SIMD instructions, Graphics hardware and software – generic architecture, I/O, GPU, Shader programming, graphics software (openGL)

Unit V **08 Hours**

GPU architectures - Streaming Multi Processors, Cache Hierarchy, The Graphics Pipeline, Introduction to CUDA programming

Unit VI **08 Hours**

Multi-dimensional mapping of dataspace, Synchronization,
Warp Scheduling, Divergence, Memory Access Coalescing

Textbooks

1. Hearn, Baker – " Computer Graphics (C version 2nd Ed.)" – Pearson education
2. F. S. Hill, Stephen Kelly, Computer Graphics using OpenGL, PHI Learning
3. David F. Rogers - Procedural Elements of Computer Graphics, Tata McGRAw Hill

Reference Books

1. Samit Bhattacharya. (2015). Computer Graphics. Oxford University Press.
2. Hearn, D. & Baker, M. P. (2003). Computer Graphics with OpenGL, (3rd ed), Pearson

List of Laboratory Exercises

1. State and discuss on basics of computer graphics.
2. Design and apply the Bresenham's circle & line drawing algorithm using C language.
3. Design and build the DDA circle & line drawing algorithm using C language
4. Design and Illustrate 2D & 3D transformation.
5. Illustrate and construct 3D clipping algorithms.
6. Build shading algorithm using OpenGL.

Project Based Learning

1. Helicopter gam
2. Sinking Ship
3. Arrival and departure of the train
4. Scientific calculator
5. Clock
6. Tower of Hanoi
7. Windmill
8. Steam engine
9. Traffic signal
10. Solar system

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Quantitative Techniques, Communication and Values

TEACHING SCHEME:	EXAMINATION SCHEME:	CREDITS:
Theory: 04 Hours / Week	Semester End Examination: 60 Marks Internal Assessment: 40 Marks	Credits:4

Course Pre-requisites: The students should have knowledge of

1. Basic math's and reasoning, and comprehensive ability
2. Basic knowledge of communication process, soft skills
3. Basic knowledge and idea about leaders and leadership qualities, ethics, etiquettes and values

Course Objective:

The **Quantitative Techniques, Communication and Values** aims to augment students to face the campus recruitment test and train them on applying short techniques/ tricks to solve questions of Maths, reasoning, and English in very less amount of time. The communication and values section focuses on the aspects of communication and soft skills such as grooming personality for leading team, presentation, business communication which would enable graduates to project themselves as a professional in the corporate sector and/or otherwise.

Course Outcomes: The student will be able to

1. Solve the aptitude test in the recruitment and competitive exam by applying short techniques and solve the question in less amount of time
2. Apply the short mnemonics and techniques to solve the questions of logical reasoning in the placement and competitive exam in lesser time.
3. Develop the verbal ability to communicate effectively using suitable vocabulary and proper sentence pattern
4. Understand the concept of soft skills and its implication at workplace
5. Build up the ability to study employment business correspondences and its proper implications
6. Understand business ethics, etiquettes and values and apply them in the professional ventures.

Course Content:

Unit-I	**QUANTITATIVE APTITUDE**: Number system, Percentage, profit and loss, Simple Interest and Compound Interest, Ratio, Proportion and Average, Mixture and Allegation, Time, Speed & Distance, Time & Work , Permutation & Combination, Probability, Pipes and Cisterns	**(8 Hrs)**
Unit-II	**NON-VERBAL REASONING:** Coding, Decoding, Number series, Blood relation Directions, cubes & dices, Data Interpretation, Data Sufficiency, Set Theory & Syllogisms, Matching, Selection & Arrangement, Clocks & Calendars, Visual Reasoning, Input, Output & Flow Chart.	**(8 Hrs)**
Unit-III	**VERBAL REASONING:** Sentence Patterns, Sentence correction and spotting errors, Vocabulary, antonyms and synonyms and analogy, Phrasal Verbs, idiomatic expressions, reading comprehension, closest, sentence rearrangement and theme detection	**(8 Hrs)**
Unit-IV	**SELF AWARENESS AND SOFT SKILLS DEVELOPMENT:** Concept of SWOT, Importance of SWOT, Individual & Organizational SWOT Analysis, Soft skills, meaning, need and importance, difference between soft skills and hard skills, life skills and personal skills, Leadership skills,-Importance ,Types, Attributes of good leader Motivational theories and leadership ,Emotional intelligence in personal and professional lives its importance need and application, Team Building and conflict resolution Skills ,Problem solving skills, Time Management and Stress Management Skills Pareto Principle(80/20) Rule in time management, Time management matrix, creativity and result orientation, working under pressure, stress management	**(8 Hrs)**
Unit-V	**COMMUNICATION AND HONING EMPLOYMENT SKILLS:** Communication process, Non-verbal codes in communication, importance of LSRW in communication, Barriers to communication, Principles of effective Technical writing, Email writing and Netiquettes, **Letter writing** – formal letters, job application letter,	**(8 Hrs)**

cover letter, structure of technical report writing, Building Resume and CV, Tips to build an effective Resume Group discussion, Skills required for Group Discussion Interview skills, Ways of handling telephonic interviews, Importance of body language, grooming &etiquettes for getting right impression in PI&GD , Extempore, Introduction to PowerPoint presentation, ,Structure & flow of presentation,

Unit-VI **BUSINESS ETHICS, ETIQUETTES AND VALUES:** **(8 Hrs)**

The Importance of Ethics and Values in Business World, Respect for Individuality and diversity at workplace values of a good manager Key features of corporate etiquette, corporate grooming & dressing, etiquettes in social & office Setting-Understand the importance of professional behaviour at the work place, Corporate social responsibility (CSR) its importance and need.

Internal Assessment:

Unit Test -1	UNIT – I, II, III
Unit Test -2	UNIT – IV, V, VI

Reference Books:

1 Quantitative Aptitude by R. S. Agarwal published by S. Chand
2 The Book of Numbers by Shakuntala Devi
3 A Modern Approach To Logical Reasoning by R. S. Agarwal published by S. Chand
4 A New Approach to Reasoning Verbal & Non-Verbal by Indu Sijwali
5 Business Communication by Meenakshi Raman, Prakash Singh published by Oxford University press, second edition
6 Communication Skills by Sanjay Kumar, Pushp Lata, published by Oxford University press, second edition
7 Technical Communication by Meenakshi Raman, Sangeeta Sharma published by Oxford University press
8 Developing Communication Skills by Krishna Mohan, Meera Banerji published by Macmillan India Pvt Ltd
9 Soft Skills by Meenkashi Raman, published by Cengage publishers
10 Soft Skills by Dr. K Alex published by Oxford University press
11 Soft skills for Managers by Dr. T. Kalyana Chakravarthi and Dr. T. Latha Chakravarthi published by biztantra

Project Based Learning Topics:

1 Prepare mock Tests on Unit –I and solve it in given time (use of PSD lab manual)
2 Prepare mock Tests on Unit –I and solve it in given time (use of PSD lab manual)

3 Prepare online model test based on Unit-II and solve it in specific time (use of PSD lab manual)
4 Prepare online model test based on Unit-II and solve it in specific time (use of PSD lab manual)
5 Form a model for spoken and written communication skills which avoid grammar mistakes and common errors
6 Develop various activity models for enriching and developing vocabulary
7 Preparing strategies by using SWOT and TWOS analysis
8 Analysing differences between Soft Skills, Hard skills, and Personal skills
9 Develop Bruce Tuchman's Team Building Models with classmates/Teammates
10 To study different personalities of Leaders from various sectors and find out their attributes and success stories
11 Preparing a model for Time Management Skills and Stress Management and conduct activities for effective implementation of it.
12 Form a model to develop LSRW and communication Skills
13 Conduct mock interview and practice GD activities to build competencies for actual selection process
14 Preparing a model for evaluating Values and Ethics of Good Managers
15 Preparing a model of dress codes and attire for different professional situations corporate etiquettes and its implications
16 Develop some good activities to understand the importance and need of Corporate social responsibility (CSR)

Mobile Architecture and Programming

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	04 Hours/Week	University Examination:	60 Marks		
Practical:	02 Hours/Week	Internal Assessment:	40 Marks	Lecture	04
		Term Work	25 Marks	Practical	01
		Practical	25 Marks		
		Total	150 Marks	Total	05

Course Objective:

1. To study different Mobile OS
2. To study practical applications of Mobile.
3. To develop problem solving abilities using Mobile Applications

Prerequisite:

Knowledge of Object-Oriented Programming, Fundamentals of Networking and Wireless Communication

Course Outcomes: On completion of the course, students will have the ability to:

1. Understand the basic concept of mobile computing
2. Explain the basic of mobile telecommunication system
3. Understand mechanisms of mobile IP to support mobility.
4. Understand mechanisms within transport layer while using TCP for moility.
5. Develop mobile application using mobile application development frameworks for Android
6. To develop M-commerce app and maintain its security.

Unit I Detailed Introduction of Mobile Computing: **08 Hours**

History, Types, Benefits, Application, Evolution, Security Concern regarding Mobile Computing, Different Propagation Modes, Wireless Architecture and its types, needs of mobile user, Mobile Development Importance, Survey of mobile based application development.

Unit II Mobile Telecommunication System: **08 Hours**

Introduction to Cellular Systems GSM Services & Architecture Protocols Connection Establishment Frequency Allocation Routing Mobility Management Security GPRS-UMTS Architecture Handover Security

Unit III Mobile IP: **08 Hours**

Need of mobile IP, IP packet delivery, Agent Discovery, Registration, Tunnelling and encapsulation, Route optimization, IP Handoff

Unit IV Mobile Transport Layer: **08 Hours**

Overview of Traditional TCP and implications of mobility control. Improvement of TCP: Indirect TCP, Snoop TCP, Mobile TCP, Fast Retransmit/fast recovery, Time-out freezing, Selective retransmission, Transaction-oriented TCP.

Unit V Mobile Operating System & Application: **08 Hours**

Mobile device operating systems, special constrains & requirements, commercial mobile operating system, software development kit, iOS, iOS architecture, Android, Android architecture, How to develop application via Android, compiling & executing programs in Android, blackberry, window phone.

Unit VI Mobile Payment System using M commerce: **08 Hours**

Mcommerce- structure, pros & cons, mobile payment system, security issues, application issues, GSM mobility management administration.

Textbooks

1. Jeff McWherter, Scott Gowell, Professional Mobile Application Development, John Wiley & Sons, Ref: www.it-ebooks.org
2. Maximiliano Firtman, Programming the mobile Web, Oreilly, 2nd Edition, 2013, ISBN: 978- 1-449-33497-0

Reference Books

1. Prashant Kumar Patnaik, Rajib mall , "Fundamental of mobile computing", PHI learning Pvt. Ltd.,2012
2. Jochen h schiller, "Mobile Communication", second edition Pearson education 2007

List of Laboratory Exercises

1. Create an application to print the multiplication table of any number entered by the user.
2. Create a student registration form (Use 'hints' instead of 'Labels').
3. Create an application that makes use of Shared Preferences to save and retrieve data in form of key-value pairs.
4. Create an application that creates a notification after clicking a button.
5. Create an application to take temperature as input from the user in degrees Celsius(°C) and display it in Fahrenheit(°F).
6. Create an application using linear Layout(Horizontal/Vertical) enabling scrollable Images/Text.
7. Create an application that will display toast (Message) on specific interval of Time.
8. Create an application using list View to display a scrollable grocery shopping list.
9. Create a Multi-screen application using intents.
10. Creating Splash screen using handler in Android

Project Based Learning

1. Create an application to making use of SQLite database for CRUD operations.
2. Create an application to play a song either from the device memory or from the internet.
3. Create a calculator application.
4. Create an application to change colour of the screen, based on selected options from the menu.
5. Create an application to open the user's favourite website/URL.
6. Women's Security with SMS Alert based android app
7. On Road Vehicle Breakdown Help Assistance
8. Location Based Garbage Management System for Smart City
9. Android based attendance monitoring system
10. Online Book Store: Ecommerce Application

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

B.Tech.(Computer Engineering)

Programme Curriculum (2021 Course)

Semester- VII

Artificial Intelligence

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
	Hours/Week		Marks		Credits
Lecture:	3 Hours/Week	**University Examination**	**60 Marks**	**Theory**	03
Practical:	2 Hours/Week			**Practical**	01
		Internal Assessment	**40 Marks**		
		Term Work	**25 Marks**		
		Practical	**25 Marks**		
		Total	**150 Marks**	**Total**	04

Course Objective:
To make student aware of basics of Artificial Intelligence (AI), Knowledge representation methods, learning concept and basics of artificial neural network.

Prerequisite:
Data structures, Algorithmic Strategies, Discrete Mathematics

Course Outcomes: On completion of the course, students will have the ability to:

7. Describe the challenges in developing AI based systems
8. Apply appropriate problem-solving strategy to solve a particular problem
9. Use appropriate knowledge representation method
10. Describe components of planning system
11. Apply the various knowledge representation strategies
12. Describe the plan generation systems

Unit I: Artificial Intelligence and its Issues **06 Hours**

Definitions - Importance of AI, Evolution of AI - Applications of AI, Classification of AI systems with respect to environment, Knowledge Inferring systems and Planning, Uncertainty and towards Learning Systems.

Unit II: Problem Solving and Search Strategies **06 Hours**

Problem solving by Search, Problem space - State space, Blind Search - Types, Performance measurement. Breadth First Search, Depth First Search, Depth Limited Search, Iterative Deepening Depth First Search, Bidirectional Search ,Comparison of Uninformed search Strategies, Searching with partial information,Sensor-lessproblems,Generate&test,HillClimbing,BestFirstSearch,A*andAO*Algorithm,Constraintsatisfaction,Game playing Minimax Search, Alpha-Beta Cutoffs, Waitingf or Quiescence

Unit III Knowledge Representation and Reasoning **06 Hours**

Logical systems Knowledge Based systems, Propositional Logic Constraints, Predicate Logic First Order Logic, Inference in First Order Logic, Ontological Representations and applications, Overview Definition of uncertainty, Bayes Rule Inference, Belief Network, Utility Based System, Decision Network.

Unit IV Planning **06 Hours**

Basic plan generation systems, Components of planning system Advanced, Strips plan generation systems, K-strips strategic explanation Planning with state-space search – partial-order planning – planning graphs – planning and acting in the real world

Unit V : Learning **06 Hours**

Learning: Learning concept, Supervised and unsupervised learning, Learning from observation - Inductive learning – Decision trees – Explanation based learning – Statistical Learning methods - Reinforcement Learning

06 Hours

Unit VI :

basic, comparison of human brain and machine, biological neuron, general neuron model, activation functions, Perceptronlearningrule,applicationsandadvantagesofneuralnetworks

Textbooks

1. Deepak Khemani , “Artifitial Intelligence”, Tata Mc Graw Hill Education ,2013
2. G. Luger, “Artificial Intelligence: Structures and Strategies for complex problem solving”, Fourth Edition, Pearson Education

Reference Books

J. Nilsson, “Artificial Intelligence: A new Synthesis”, Elsevier Publishers.

List of Practical Experiments

1. Apply A* and A* algorithm and solve a search problem.
2. Write a program to conduct uninformed and informed search
3. Represent the knowledge for the given problem
4. Write a program to solve water-jug problem.
5. Write a menu driven program to display set of questions to user and give answer of selected question
   .For Ex.Menu will have following set of questions
   i). Who likes apple?
   ii) Does anybody like apple?
   iii) Is it true that nobody likes apple?
   iv) Who likes apple as well as enjoys playing cricket and piano?
   v) Does anybody play at least one instrument?
   vi) Who likes to play chess, drink buttermilk but does not play any instrument?
   vii) Who share at least one hobby and at least one instrument?
   viii) Who are the persons sharing common instruments but no hobbies are in common?
   Ask the user to enter the question no. The programs should display the answer.

6 Write a program to solve travelling salesman problem.
7 Write a program to solve monkey-banana problem
8. Write a program to solve tower of Hanoi problem.
9 Write a program to solve Tic-Tac-Toe problem.
10 Write a program to solve 8-puzzle problem.

Project Based Learning - Provisional List of Projects

1. Handwritten digits recognition.
2. Lane line detection.
3. Spam classifier
4. Optimal path finder
5. Fire detection and localization using surveillance.

6. Next character predictor
7. Chatbot
8. Game development
9. Recommender system
10. Face recognition
11. Dog and Cat classification

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Virtualisation and Cloud Computing

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
	Hours/Week		**Marks**		**Credits**
Lecture:	3 Hours/Week	**University Examination**	**60 Marks**	**Theory**	03
Practical:	2 Hours/Week			**Practical**	01
		Internal Assessment	**40 Marks**		
		Term Work	**25 Marks**		
		Oral	**25 Marks**		
		Total	**150 Marks**	**Total**	04

Course Objective:

To understand the various distributed system models and evolving computing paradigms
To gain knowledge in virtualization of computer resources
To realize the reasons for migrating into cloud
To introduce the various levels of services that can be achieved by a cloud.
To describe the security aspects in cloud and the services offered by a cloud.

Prerequisite: Basic knowledge Operating System

Course Outcomes: On completion of the course, students will have the ability to:

1. Ability to understand various service delivery models of a cloud computing
2. Ability to understand the ways in which the cloud can be programmed and deployed.
3. Ability to understand the virtualization and cloud computing concepts
4. Assess the comparative advantages and disadvantages of Virtualization technology
5. Analyse authentication, confidentiality, and privacy issues in cloud computing
6. Identify security implications in cloud computing

Unit I: Cloud Computing Fundamentals **06 Hours**

Cloud Computing Fundamentals: Definition of Cloud computing, Roots of Cloud Computing, Layers and Types of Clouds, Desired Features of a Cloud, Cloud Infrastructure Management, Infrastructure as a Service Providers, Platform as a Service Providers. Computing Paradigms: High-Performance Computing, Parallel Computing, Distributed Computing, Cluster Computing, Grid Computing.

Unit II: Virtualization **06 Hours**

Virtualization: Introduction to Cloud Computing- Cloud issues and challenges – Properties – Characteristics – Service models, Deployment models. Cloud resources: Network and API – Virtual and Physical computational resources – Data-storage. Virtualization concepts – Types of Virtualizations- Introduction to Various Hypervisors – High Availability (HA)/Disaster Recovery (DR) using Virtualization, Moving VMs.

Unit III: Service Models **06 Hours**

Service Models: Infrastructure as a Service (IaaS) – Resource Virtualization: Server, Storage, Network – Case studies. Platform as a Service (PaaS) – Cloud platform & Management: Computation, Storage – Case studies. Software as a

Service (SaaS) – Web services – Web 2.0 – Web OS – Case studies – Anything as a service (XaaS) – Microservices.

Unit IV: Cloud Programming and Software Environments **06 Hours**

Cloud Programming and Software Environments: Cloud Programming and Software Environments – Parallel and Distributed Programming paradigms – Current technologies – Programming support of App Engines – Emerging Cloud software Environment.

Unit V: Cloud Access **06 Hours**

Cloud Access: authentication, authorization and accounting – Cloud Provenance and meta-data – Cloud Reliability and fault-tolerance – Cloud Security, privacy, policy and compliance- Cloud federation, interoperability and standards.

06 Hours

Unit VI: Cloud Technologies And Advancements And SLA Management

Hadoop – MapReduce – Virtual Box -- Google App Engine – Programming Environment for Google App Engine – Open Stack – Federation in the Cloud – Four Levels of Federation – Federated Services and Applications – Future of Federation.

SLA Management in cloud computing: Traditional Approaches to SLO Management, Types of SLA, Life Cycle of SLA, SLA Management in Cloud.

Textbooks

1. Cloud Computing Principles and Paradigms, by Rajkumar Buyya.
2. Essentials of cloud Computing: K. Chandrasekhran, CRC press, 2014
3. Michael Miller, Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online, Que Publishing, August 2008.
4. Cloud Computing, A Practical Approach, Anthony T Velte, Toby J Velte, Robert Elsenpeter, TMH

Reference Books

1. Cloud Computing: A Practical Approach, Anthony T.Velte, Toby J.Velte, Robert Elsenpeter, Tata McGraw Hill,rp2011.
2. Cloud Application Architectures: Building Applications and Infrastructure in the Cloud, George Reese, O'reilly, SPD,rp2011.
3. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, Subra Kumaraswamy, Shahed Latif, O'Reilly, SPD, rp2011

List of Assignments

1. Installation and configuration of own Cloud.
2. Implementation of Virtualization in Cloud Computing to Learn Virtualization Basics, Benefits of Virtualization in Cloud using Open Source Operating System.
3. Study and implementation of infrastructure as Service using Open Stack.
4. Write a program for Web feed using PHP and HTML.
5. Write a Program to Create, Manage and groups User accounts in own Cloud by Installing Administrative Features.
6. Design and develop custom Application using Salesforce Cloud
7. creating an AMI for Hadoop and implementing short Hadoop programs on the Amazon Web Services platform.
8. Creating an Application in SalesForce.com using Apex programming Language

9. Design an Assignment to retrieve, verify, and store user credentials using Firebase Authentication, the Google App Engine standard environment, and Google Cloud Data store.

Project Based Learning - Provisional List of Projects

1. Data Science Assignment Help in Microsoft Azure Specify the necessary environment as a Docker file.
2. Cloud based VM resources for application hosting
3. Configurable deployment of cloud applications using the Docker container
4. Big Data analytics on unstructured text data using Microsoft Azure.
5. Hadoop and MapReduce in Microsoft HDInsight.
6. Azure Machine Learning for sentiment analysis
7. Cloud Computing Mashup/Docker Project
8. Deployment to a publicly hosted Linux VM [Azure or AWS will be appropriate here.]

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Scalable Computing

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
	Hours/Week		**Marks**		**Credits**
Lecture:	3 Hours/Week	**University Examination**	**60 Marks**	**Theory**	03
Practical:	2 Hours/Week			**Practical**	01
		Internal Assessment	**40 Marks**		
		Term Work	**25 Marks**		
		Oral	**25 Marks**		
		Total	**150 Marks**	**Total**	04

Course Objective: - This course covers computer systems topics that are essential for students engaging in computational and data sciences. It introduces topics on architecture, OS and data structures. Learn advanced topics on tree/graph data structures, HPC/GPGPU programming and Big Data platforms.

Prerequisite: Basic knowledge of computer systems, data structures and programming, and algorithms. However, the course will have a rapid pace and students are expected to pick up the skills rapidly through self-learning.

Course Outcomes: On completion of the course, students will have the ability to:

1. Understand challenges in efficient execution of large-scale parallel applications
2. Identify Big Data and its Business Implications
3. Understand Grid computing services and its practical use
4. Understand data center network and parallel code analysis
5. Identify techniques for self-updating
6. To have knowledge of the fundamentals of Green Computing.

Unit I : Parallel Computing **06 Hours**

Why parallel computing? Shared memory and distributed memory parallelism, Amdahl's law, speedup and efficiency, supercomputers.
Scalability, benchmarking, performance modeling, impact of network topologies, parallel code analysis and profiling.

Unit II : Big data Computing **06 Hours**

Types of Big Data, Design goals of Big Data platforms, and where in the systems landscape these platforms fall. Distributed programming models for Big Data, including Map Reduce, Stream processing and Graph processing. Runtime Systems for Big Data platforms and their optimizations on commodity clusters and Clouds. Scaling data Science algorithms and analytics using Big Data platforms.

Unit III : Grid Computing **06 Hours**

Introduction to Open Grid Services Architecture (OGSA) – Motivation – Functionality Requirements – Practical & Detailed view of OGSA/OGSI – Data intensive grid service models – OGSA services.

Unit IV: Data Center Computing **06 Hours**

Introduction of data centre, Core components of data centre, data centre infrastructure, types of data centre, Key parameters of data centre, data center network designs, software-defined networks (SDN), virtualization technologies, data center security, traffic engineering, resource management, and green data centers.

Unit V: Automatic Computing **06 Hours**

Autonomic Computing: Objective, element of automatic computing- Self-organizing network. Self-healing systems. Self-optimization. Characteristic of automated computing, component of computing, four areas of automatic computing (self-configuration, self-healing (error correction), self-optimization (automatic resource control for optimal functioning) and self-protection (identification and protection from attacks in a proactive manner) defined by IBM. Example of Automatic Computing.

Unit VI: Green Computing **06 Hours**

Green IT Fundamentals: Business, IT, and the Environment – Green computing: carbon foot print, scoop on power – Green IT Strategies: Drivers, Dimensions, and Goals – Environmentally Responsible Business: Policies, Practices, and Metrics.

Textbooks

1. Data-Intensive Text Processing with MapReduce, Jimmy Lin and Chris Dyer, 1st Edition, Morgan & Claypool Publishers, 2010
2. Kai Hwang, Geoffery C. Fox and Jack J. Dongarra, "Distributed and Cloud Computing: Clusters, Grids, Clouds and the Future of Internet", First Edition, Morgan Kaufman Publisher, an Imprint of Elsevier, 2012.
3. Bhuvan Unhelkar, ―Green IT Strategies and Applications-Using Environmental Intelligence‖, CRC Press, June 2014.
4. Peter S Pacheco, An Introduction to Parallel Programming, Morgan Kaufmann, 2011.

Reference Books

1. Dirk Deroos et al., Hadoop for Dummies, Dreamtech Press, 2014.
2. JL Hennessy and DA Patterson, Computer Architecture: A Quantitative Approach, 4th Ed., Morgan Kaufmann/Els India, 2006.
3. Alin Gales, Michael Schaefer, Mike Ebbers, ―Green Data Center: steps for the Journey‖, Shroff/IBM rebook, 2011.

List of Assignments

1. Basics of MPI (Message Passing Interface)
2. Study of MPI collective operations using 'Synchronization'
3. Installation of Hadoop: File Management tasks in Hadoop
4. Case study on Task Assignment for grid computing.
5. Google's Green Data Centers: Network POP Case Study
6. The Case for Automated Planning in Autonomic Computing

7. Green Computing case study: calls for proposing solutions for the Arabian Gulf Oil Company.

Project Based Learning:

1. Applied Parallel Programming Languages
2. Network simulator for parallel computing
3. Big Data for cybersecurity.
4. Health status prediction.
5. Anomaly detection in cloud servers.
6. Recruitment for Big Data job profiles
7. Sound-Based Computer Automation Using Python
8. Home Automation System Using a Simple Android App

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Elective I: Software Testing & Quality Assurance

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
	Hours/Week		Marks		Credits
Lecture:	4 Hours/Week	University Examination	60 Marks	Theory	04
Tutorial:	1 Hours/Week			Tutorial	01
		Internal Assessment	40 Marks		
		Total	100 Marks	Total	05

Course Objective:
Students will learn: Introduce basic concepts of software testing.
Understand white box, block box, object oriented, web based and cloud testing.
Know in details automation testing and tools used for automation testing.
Understand the importance of software quality and assurance software systems development.

Prerequisite:
Programming Language, DBMS, JavaScript and HTML 5.

Course Outcomes: On completion of the course, students will have the ability to:

7. Understand fundamental concepts in software testing such as manual testing,
8. Design and Develop project test plan, design test cases, test data, and conduct test
9. Apply recent automation tool for various software testing for testing software.
10. Understand fundamental concepts of Selenium IDE, Selenium RC, Selenium Web driver, Selenium Grid.
11. Apply different approaches of quality management, assurance, and quality standard.
12. Apply and Analyze effectiveness Software Quality Tools

Unit I: Testing methodology **08 Hours**
Goals of Software Testing, Software Testing 10 Methodology Definitions, Model for Software Testing, Effective Software Testing vs Exhaustive Software Testing, Software Failure Case Studies, Software Testing Terminology, Software Testing Life Cycle (STLC), Software Testing methodology, Verification and Validation, Verification requirements, Verification of high level design, Verification of low level design, validation.

Unit II: Testing techniques **08 Hours**
Dynamic Testing: Black Box testing: boundary value analysis, equivalence class testing, state table based testing, cause-effect graphing based testing, error guessing. White box Testing Techniques: logic coverage criteria, basis path testing, graph matrices, loop testing, data flow testing, mutation testing. Static Testing. Validation Activities: Unit validation, Integration, Function, System, Acceptance Testing. Regression Testing: Progressive vs. Regressive, regression testing produces quality software, regression testability, objectives of regression testing, regression testing types, define problem, regression testing techniques.

Unit III : Test Management **08 Hours**
Test organization, Test planning, Detailed Test design and Test Process specification. Software Metrics: need, definition and classification of software matrices. Testing Metrics for Monitoring and Controlling the Testing Process:

attributes and corresponding matrics, estimation model for **testing effort**, information flow matrix used for testing, function point and test point. Efficient Test Suite Management, test suite prioritization its type, techniques and measuring effectiveness.

Unit IV: Test Automation and Testing Tools **08 Hours**

Need, categorization, 8 Automation selection and cost in testing tool, guidelines for testing tools. Study of testing tools: WinRunner, QTP, LoadRunner, Test Director and IBM Rational Functional Tester, Selenium etc.

Unit V : Testing Object Oriented Software **08 Hours**

OOT basics, Object- 5 for Specialized oriented testing. Environment Testing Web based Systems: Web based system, web technology evaluation, traditional software and web-based software, challenges in testing for web-based software, testing web-based testing, Testing a data warehouse.

08 Hours

Unit VI : Quality Software Quality Management

McCall's quality factors 3 Management and Criteria, ISO 9126 quality characteristics, ISO 9000:2000, software quality management.

Textbooks

5. William E. Lewis" Software Testing and Continuous Quality Improvement, CRC Press.
6. Dorothy Graham, Erik van Veenendaal, Isabel Evans, Rex Black, Foundations of Software Testing, Cengage Learning.

Reference Books

4. M. G. Limaye, Software Testing: Principles, Techniques and Tools, TMH
5. Paul C. Jorgenson, Software Testing: A Craftsman's Approach, CRC Press

Project Based Learning - Provisional List of Projects

1. Customer Experience Management
2. Android Local Train Ticketing System
3. Android Task Monitoring
4. Android Women Safety App
5. Personality Analysis
6. Online Election System
7. Analyzing sentiments of Facebook Users: A Software System
8. Detecting Evil Applications on Online Social Networks.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Elective I: Mobile Operating System

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
	Hours/Week		**Marks**		**Credits**
Lecture:	4 Hours/Week	**University Examination**	**60 Marks**	**Theory**	04
Tutorial:	1 Hours/Week			**Practical**	01
		Internal Assessment	**40 Marks**		
		Total	**100 Marks**	**Total**	05

Course Objective: To make students aware of mobile operating system, framework of mobile operating system, security aspects.

Prerequisite: Basic knowledge of Operating System.

Course Outcomes: On completion of the course, students will have the ability to:

1. To study the basic concept of mobile operating system.
2. To implement the concept of multiprogramming.
3. To gain the knowledge of digital certificate and Security.
4. To understand the framework of mobile OS.
5. Learn the concepts of Linux OS.
6. To implement case study

Unit I: Introduction to Mobile Operating Systems **08 Hours**

Brief History of Mobile Operating Systems, OS-Interfaces, Multilevel Views of OS, Categories, Small and Specialized OS, 64-Bit OS, Processes and Threads, System Performance and Models: Performance of Computer Systems, Performance Metrics, Workload and System Parameters, Simulation Models: Types, Discrete-Event Model, Stochastic Model.

Unit II: Multiprogramming **08 Hours**

System with Multiprogramming, Processor Scheduling, Synchronization, Deadlocks, File Management, Memory Management: Process Address Space, Contiguous Memory Allocation, Non Contiguous Memory Allocation, Virtual Memory, Paging with Virtual Memory.

Unit III: Security and Protection **08 Hours**

Components for Security and Protection, Physical Security, User Authentication, Protection, Secure Communications, Digital Certificates, System Vulnerabilities, Invasive and Malicious Software, Defending the System and User, Intrusion Detection Management.

Unit IV: Mobile Ecosystems **08 Hours**

Application Framework, Developing a Mobile Strategy, Mobile Information Architecture, Mobile Design: Elements of Mobile Design, Ubiquity in the Mobile Web, Mobile Web Development.

Unit V: Introduction to Linux **08 Hours**

Command Line Interface, Files and Directories, Shell Variables, Script Files, Connecting a Remote Linux Server. Java Modelling Framework, Java and Posix Threads.

Unit VI: Case Study **08 Hours**

Android SDK, iOS, Windows, Mobile Web Apps vs. Mobile Applications

Textbooks:

1. Jose M Garrido, Richard Schlesinger, Kenneth Hoganson, Principles of Modern Operating Systems.
2. By Brian Fling, Mobile Design and Development: Practical concepts and techniques for Creating Mobile Sites and Web Apps, O'Reilly Publications.

Reference Books:

Mobile Operating Systems, Gerard Blokdyk

Project Based Learning:

1.Understand the file system of Linux.
2.Implement Shell Scripting of Linux.
3. Bus pass management system using Android Studio.
4.Online Book Store: Ecommerce Application.
5. Agri Shop For Farmers Online Shopping Android Application.
6. COVID-19 (corona) Online Test Results & availability booking of Hospitals based Mobile App.
7. Online Voting System Project Application.
8. On Road Vehicle Breakdown Assistance (ORVBA) Finder Project

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Elective I: Fundamentals of Fog & Edge Computing

TEACHING SCHEME		**EXAMINATION SCHEME**		**CREDIT SCHEME**	
	Hours/Week		**Marks**		**Credits**
Lecture:	4 Hours/Week	**University Examination**	**60 Marks**	**Lecture:**	04
Tutorial:	1 Hours/Week			**Tutorial:**	01
		Internal Assessment	**40 Marks**		
		Total	**100 Marks**		05

Course Objective:

1.Introduce cloud computing and enabling technologies

2.Explore the need for fog and edge computation

3.Impart the knowledge to log the sensor data and to perform further data analytics

Prerequisite:

1. Principles of Cloud Computing
2. Python Programming
3. Java programming

Course Outcomes:

1. Understand the principles, architectures of fog
2. Understand the communication and management of fogs
3. Understand storage and computation in fogs
4. Design and Implement Internet of Everything (IoE) applications through fog computing architecture computing
5. Analysis the performance of the applications developed using fog architecture
6. Understand the security and privacy issues of fog computing

Unit I: Internet of Things (IoT) and New Computing Paradigms — **08 Hours**

Introduction-Relevant Technologies-Fog and Edge Computing Completing the Cloud-Hierarchy of Fog and Edge Computing-Business Models-Opportunities and Challenges

Unit II: Challenges in Federating Edge Resources — **08 Hours**

Introduction-Methodology-Integrated C2F2T Literature by Modeling Technique-Integrated C2F2T Literature by Use-Case Scenarios-Integrated C2F2T Literature by Metrics-Future Research Directions

Unit III: Management and Orchestration of Network Slices in 5G, Fog, Edge, and Clouds — **08 Hours**

Introduction-Background-Network Slicing-Network Slicing in Software-Defined Clouds-Network Slicing Management in Edge and Fog- Internet of Vehicles: Architecture, Protocol and Security Seven layered model architecture for Internet of Vehicles- IoV: Network Models, Challenges and future aspects.

Unit IV: Optimization Problems in Fog and Edge Computing — **08 Hours**

Preliminaries-The Case for Optimization in Fog Computing-Formal Modeling Framework for Fog Computing-Metrics-Further Quality Attributes-Optimization Opportunities along the Fog

Architecture-Optimization Opportunities along the Service Life Cycle-Toward a Taxonomy of Optimization Problems in Fog Computing.

Unit V: Middleware for Fog and Edge Computing: Design Issues **08 Hours**

Need for Fog and Edge Computing Middleware-Design Goals-State-of-the-Art Middleware Infrastructures-System Model-Proposed Architecture-Case Study Example

Unit VI: Technologies in Fog Computing and Applications **08 Hours**

Fog Data Management-Motivating Example: Smart Building-Predictive Analysis with Fog Torch Machine Learning in Fog Computing-Data Analytics in the Fog-Data Analytics in the Fog Architecture-Configurations Tracking-Fog Computing Model for Evolving Smart Transportation Applications-Testing Perspectives of Fog-Based IoT Applications-Legal Aspects of Operating IoT Applications in the Fog

Textbooks:

1. Buyya, Rajkumar, and Satish Narayana Srirama, eds, Fog and edge computing: principles and paradigms, 1st edition, John Wiley & Sons, 2019.
2. John Mutumba Bilay , Peter Gutsche, Mandy Krimmel and Volker Stiehl ,SAP Cloud Platform Integration: The Comprehensive Guide, 2nd edition, Rheinwerg publishing, 2019

Reference Books

1. Bahga, Arshdeep, and Vijay Madisetti. Cloud computing: A hands-on approach, 1st edition, CreateSpace Independent Publishing Platform, 2013.
2. Ovidiu Vermesan, Peter Friess, Internet of Things –From Research and Innovation to Market Deployment, 1 st edition,River Publishers, 2014
3. Michael Missbach, Thorsten Staerk, Cameron Gardiner, Joshua McCloud, Robert Madl, Mark Tempes, George Anderson, SAP on Cloud, 1 st edition, Springer, 2016

Project Based Learning:

1. Scheduling for Deep Reinforcement Learning-Based Offloading in Vehicle Edge Computing
2. Multilevel vehicular edge-cloud computing networks with advanced deep learning-based computational offloading
3. In Wireless Metro Area Networks, Optimal Cloudlet Location and User to Cloudlet Allocation
4. Joint Management and Cloud Unloading for Mobile Applications at the Optimal Level
5. Mobile Cloud Computing: Distributed Mega Pricing for Effective Application Offloading
6. An Edge Node Resource Management Framework
7. Increasing the Reliability of Cloud Services by Using a Proactive Fault-Tolerance Approach
8. Task assignment for mobile edge computing that considers user mobility
9. Deadline-Aware Portable Edge Computing Systems Task Scheduling
10. A Privacy-Preserving Data Gathering Scheme for IoT Applications Assisted by Mobile Edge Computing

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Elective I: System Thinking

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
	Hours/Week		**Marks**		**Credits**
Lecture:	4 Hours/Week	**University Examination**	**60 Marks**	**Lecture:**	04
Tutorial:	1 Hours/Week			**Tutorial:**	01
		Internal Assessment	**40 Marks**		
		Total	**100 Marks**		05

Course Objective:
Systems thinking is a holistic approach to analysis that focuses on the way that a system's constituent parts interrelate and how systems work over time and within the context of larger systems.

Prerequisite: Basic knowledge of System Thinking

Course Outcomes: On completion of the course, students will have the ability to:

1. Provide an overview of the history, research and perspectives into systems thinking
2. Understand and document system thinking objectives
3. Establish a basic understanding of systems thinking terminology, theories, processes, methods, language and tools.
4. Evaluate when it is appropriate to apply thinking methods, i.e. reductionist methods (ex. data collection, scientific method, etc.) as opposed to applying systems thinking methods (ex. ,Systems Engineering, Breakthrough Thinking/Smart Questions, etc.)
5. Describe and model solutions that will enable system thinking ex. (mind maps, feedback & causal loops, behaviour over time diagrams
6. Apply systems engineering and analysis techniques to various problems. (socio - technical, supply chain, value chain / lean, etc.)

Unit 1 - Introduction to System Thinking **08 Hours**

Introduction: Definitions & Concepts, What is Systems Thinking?, The importance and Purpose & Principles of Systems Thinking, Systems Thinking tools and techniques, The Systems Thinking Process/Protocol for Business, Applying Systems Thinking to Engineering, System Thinking in Management.

Unit II - System Engineering, Data Analytics and System Thinking **08 Hours**

The fourth industrial revolution, Integrating Reliability Engineering with System Engineering, Software Cybernetics, Using Modeling & Simulations ,Risk Management, An Integrated Approach to Safety & Security Based on Systems Theory, Applied system Thinking, The System in System Thinking Applied System Thinking Approach, Soft System Methodology, Systemigram, Casual Loop Diagrams, Intervention Points, Approach , Tools & Methods- Final Thoughts

Unit III - System Thinking in Software Engineering (Say Agile Approach) **08 Hours**

Principles of System Thinking for Software Development, The critical role of systems thinking in software development, IT Project Management with System Thinking, Applying system thinking to model-based software engineering.

Unit IV - System Thinking in Project Management **08 Hours**

Systems Thinking for Project Management, The Need for Systems Thinking in Project Management, Systems thinking and its latent potential in project Planning, Systems thinking and its latent, potential in project implementation and control, How to Apply Systems Thinking in Managing Projects, Managing Project Risks, Improving Decisions in Projects, Systems Approaches: Hard Systems Thinking, System Dynamics, Organizational Cybernetics, Complexity Theory, Strategic Assumption Surfacing and Testing Interactive Planning, Soft Systems Methodology, Critical Systems Heuristics, Team Syntegrity, Postmodern Systems Thinking

Unit V - System Thinking for Intelligent Systems **08 Hours**

Engineering Intelligent Systems: Systems Engineering and Design with Artificial Intelligence, Visual Modelling, and Systems Thinking, Artificial Intelligence, Science Fiction, and Fear, The Intelligence in the System: How Artificial Intelligence Really Works, What Is Artificial Intelligence?, Modelling of Intelligent System Thinking in Complex Adaptive Systems, Systems Thinking and AI applications

Unit VI - System thinking future research and Case studies **08 Hours**

Systems thinking and complexity ideas in Research, Key themes and perspectives, A systematic review of the use of key, Methodology, Analysis, Results, Discussion, Limitations, Ways Forward, Conclusion.

Textbooks:

1. Engineering Intelligent Systems, Systems Engineering and Design with Artificial Intelligence, Visual Modelling, and Systems Thinking By Barclay R. Brown · 2022
2. Systems Engineering, Building Successful Systems By Howard Eisner · 2011
3. Systems Engineering in the Fourth Industrial Revolution, Big Data, Novel Technologies, and Modern Systems Engineering

Reference Books

1. The Journey to Enterprise Agility Systems Thinking and Organizational Legacy By Daryl Kulak, Hong Li · 2017
2. Mastering Project Management James P. Lewis.

Project Based Learning:

1. Understand Events, Patterns behaviour, System and Mental Model.
2. Understand and implement key concepts of System Thinking.
3. Create a list of different possible solutions. Evaluate the solutions to see whether they are realistic.
4. Conduct small tests of change to see whether an improvement can be made.
5. To understand the interrelationships within a system's structure.
6. To perform problem solving in complex system.
7. To develop the systems-thinking skills that thought leaders across the globe consider critical for 21st-century life.
8. Understand and implement different levels of systems thinking maturity.
9. Study different types of Tools for Systems Thinkers.
10. Observe events or data, to identifying patterns of behaviour overtime, to surfacing the underlying structures that drive those events and patterns.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Programming Technologies and Tools Laboratory – VI

Teaching Scheme		**Examination Scheme**		**Credit Scheme**	
	Hours/Week		**Marks**		**Credits**
Lecture:	**01 Hours/Week**	**Practical**	**50 Marks**	**Lecture**	**01**
Practical:	**02 Hours/Week**			**Practical**	**01**
		Total	**50 Marks**	**Total**	**02**

Course Objective: To develop applications in various domain by applying programming skills using Python Libraries.

Prerequisite: Understanding of basic python programming and OOPs concepts.

Course Outcomes: On completion of the course, students will have the ability to:

1. To learn how to use regular expression in Python for searching.
2. To develop the ability to write database applications in Python.
3. To develop the skill of data science using python
4. To develop the ability of Data Visualization using Python
5. To develop the skill of designing Graphical user Interfaces in Python•
6. To develop the ability to implement machine learning and deep learning applications.

Unit I : Python Regular Expressions **02 Hours**

RegEx Module, Regexes in Python and Their Uses, The match Function, Find all Function, The search Function ,Matching vs searching ,Search and Replace ,Regular Expression Modifiers

Unit II: Databases with Python **02 Hours**

Python MySQL Database access, Install the MySQL db and other Packages, **Create Database, Create table, Insert, Select, Where, Order By, Delete, Drop Table, Update,** Introduction to Python MongoDB, SQLite.

Unit III : Data Science using Python **02 Hours**

Introduction to NumPy, Installation of NumPy, Creating Arrays, **Array Indexing ,** NumPy **Data Types ,**NumPy **Array Shape ,NumPy Functions ,Basic operations on single array ,Searching and sorting.**

Unit IV: Data Visualization in Python: **02 Hours**

Pandas: Installation of Pandas, Import Pandas, Read CSV Files, Viewing the Data, Pandas -**Cleaning Data,** Removing Duplicates, Difference between Pandas and NumPy, Pandas Time Series

Python Matplotlib: Installation of Matplotlib ,Basic plotting ,Bar Plot ,Line ,Pie chart, Histogram

Unit V: GUI in Python **02 Hours**

Python Tkinter , Widgets, Python Tkinter Geometry, Python Tkinter place() method, Button ,Introduction to PyQT

Unit VI: Machine Learning using Python **02 Hours**

What is Scikit-Learn (Sklearn), Loading data, Splitting, Train Test Data, Introduction to TensorFlow

Textbooks

1. Hands-on Data Analysis and Visualization with Pandas Paperback – 1 January 2020 by Purna Chander Rao. Kathula
2. Machine learning and Deep learning using Python and Tensor flow by Venkata Reddy Konasani and Shailendra Kadre

Reference Books

- MySQL for Python: Database Access Made Easy Kindle Edition by Albert Lukaszewski
- Python GUI Programming with Tkinter, Alan D. Moore, O'Reilly Media, Inc.

List of Laboratory Exercises

- Write a program to implement algorithm that searches for an element in a list.
- Write a program to implement Databases (MySQL, MongoDB).
- Write a program to implement Calculator in Python.
- Write a program to implement GUI in Python.
- Write a program to implement simple graphs using Matplotlib in Python.
- Write a program to generate random numbers for Gaussian distribution using Numpy.

Syllabus for Unit Tests: NA

Project Stage -I

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
					Credits
				Theory	00
Practical:	2 Hrs/Week			**Practical:**	03
		Term Work :	50 Marks	**Term Work**	
		Oral :	50 Marks	**Oral:**	
		Total	100 marks	**Total**	**03**

Course Pre-requisites:
Basics of Software engineering, Software testing and knowledge of core computer engineering subjects.

Course Objectives:

- To provide in depth outline for software project planning.

Course Outcomes: On completion of the course, students will have the ability to:

7. Review and understand how previous experiences had an impact on affective states and intellectual performance.
8. Identify and define the problem.
9. Demonstrate the ability of decision-making to propose solution.
10. Design an algorithm to solve the problem.
11. Demonstrate an ability to work as a team member.
12. Perform requirement analysis process and decide feasibility.

1. The project will be undertaken preferably by a group of at least 3- 4 students who will jointly work and implement the project over the academic year. The work will involve the design of a system or subsystem in the area of Computer Engineering.
2. If the project is chosen a hardware project it will involve the designing a system or subsystem or upgrading an existing system. The design must be implemented into a working model with necessary software interfacing and a user manual.
3. If the project is chosen in the pure Software Application it must involve the detail Software Design Specifications, Data Structure Layout, File Design, Testing with complete documentation and user interface, with life cycle testing and as an executable package.
4. The group will select a project with the approval of the guide (Staff members assigned) and submit the name of the project with a synopsis of 2 or 3 pages in the month of August in the academic year. A preliminary study report by the group must be submitted and certified at the end of seventh Semester.
5. It is expected that at least one research paper is published by each group with guide.
The project report stage-I will contain the details.
Problem definition and requirement specification, acceptance test procedure (ATP).
a) System definition, requirement analysis.
b) System design with UML.
c) Documentation and references.

Documentation will use UML approach with Presentation, Category, Use Case, Class Diagrams, etc.
Project report must be checked for plagiarism from respective guide

Internship

TEACHING SCHEME	**EXAMINATION SCHEME**		**CREDIT SCHEME**	
				Credits
Practical: --			**Practical:**	03
	Termwork :	25 Marks		
	Oral	**:** 25 Marks	**Termwork**	
			Oral	
	Total	50 marks	**Total**	**03**

Course Pre-requisites:
Professional Skills, Knowledge of core computer engineering subjects.

Course Objectives:

- To provide exposure for the students on technology /tools for software development.in practical engineering fields.
- To identify their skills, values, beliefs, interests and personal abilities for professional growth.

Course Outcomes: On completion of the course, students will have the ability to:

- Propose a solution to solve real world problems with the help of technology.
- Apply software engineering principles.
- Evaluate and compare the various methodologies to solve a real-world problem.
- Demonstrate use of modern software development tools.
- Prepare and present a report on industrial training.
- Identify social and ethical responsibilities and develop skills to compete for lifelong learning.

As a part of the B. Tech Computer Engineering curriculum, Internship is a Practical course, which the students B. Tech Computer Engineering should undergo in reputed Private / Public Sector / Government organization / companies as industrial training of 60 days to be undergone by the student in the summer vacation after the semester VI. Examination and Oral examination will be conducted at the end of the semester VII

The Internship Report:

Internship report should be prepared by each student duly signed by respective guide. The report is expected to demonstrate, development of practical and professional skills in Engineering through technical experience and application of theoretical knowledge. Development of skills in dealing with people, and communication skills form part of the training experience. Students should seek advice from their employers to ensure that no confidential material is included into the report. The student should be able to present the report to prospective employers.

The following should be observed:

- Duration of Internship.
- Preliminary information
- Technical report/diary references should be made in the text to books, technical papers, standards etc., used during the training period and should be listed.
- Finally, a conclusion should include comprehensive comments on the type and value of experience gained, and how this relates to your professional career.
- A copy of the report and training certificate should be submitted to his/her employer, another copy to the Department (through the respective Adviser).
- Students should also retain a personal copy of the report.

B.Tech.(Computer Engineering)

Programme Curriculum (2021 Course)

Semester- VIII

Machine Learning

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
	Hours/Week		**Marks**		**Credits**
Lecture:	3 Hours/Week	**University Examination**	**60 Marks**	**Lecture:**	3
Practical:	2 Hours/Week			**Practical:**	1
		Internal Assessment	**40 Marks**		
		Term Work	**25 Marks**		
		Practical	**25 Marks**		
		Total	**150 Marks**	**Total**	4

Course Objectives: To provide a strong foundation of Machine Learning concepts and techniques.

Prerequisite:
The students should be aware Discrete Mathematics, Database Management System, Engineering Mathematics, Programming Languages.

Course Outcomes: On completion of the course, students will have the ability to:

1. Explain significance of Machine Learning
2. Understand different paradigms of Machine Learning.
3. Understand various machine learning models.
4. Explain supervised learning algorithms.
5. Explain unsupervised learning algorithms.
6. Tackle real world problems in the domain of machine learning, Information Retrieval.

Unit I Introduction to machine learning: **06 Hours**
Introduction to Learning Systems, Structure of Learning System, Testing vs Training, Learning vs Designing, Goal and Applications of Machine Learning, Examples of Machine Learning Problems, Need of Learning, Machine Intelligence.

Unit II Machine Learning Techniques: **06 Hours**
Types of data in Machine Learning, Structures of data, Introduction
Machine Learning Techniques: Supervised Learning (SL), Semi Supervised Learning (SSL), Unsupervised Learning (USL), Data quality and remediation, Data Pre-Processing: Dimensionality reduction, Feature subset selection.

Unit III Machine Learning Models **06 Hours**
Steps to choose Machine Learning Technique, Machine Learning Models with Examples: Linear based Models, Logic Based and Algebraic Models, Probabilistic Models.

Unit IV: Supervised Learning **06 Hours**
What is Supervised Learning? Types of Supervised Learning, Classification: What is Classification? Types: Naive Bayes Classifier, Decision Trees, Support Vector Machines, Rule based Classification, Backpropagation, Associative Classification, Classifier Accuracy Measures, Precision and Recall Measures. Regression and types Types: Linear Regression, Logistic

Regression, Classification vs Regression, Issues Regarding Classification, and Regression, Assessing performance of Regression

Unit V: Unsupervised Learning: **06 Hours**

Introduction to clustering, Types: K Means clustering Algorithm, Mixture Models, Hierarchical Clustering, Anomaly Detection, Neural Networks, Self-Organizing Map (SOM), Applications of Unsupervised Learning.

Unit VI: Trends in Machine Learning **06 Hours**

Ensemble methods for increasing accuracy: Bagging and Boosting, multitask learning, online learning and Sequence prediction, Data Streams and Active Learning, Introduction to Deep Learning and Reinforcement Learning, Case Study: Latest Machine Learning Tools.

Textbooks

1. Machine Learning (McGraw-Hill International Editions Computer Science Series
2. Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow
3. Machine Learning: A First Course for Engineers and Scientists, by Andreas Lindholm (Author), Niklas Wahlström (Author), Fredrik Lindsten (Author)

Reference Books

1. T.M. Mitchell, "Machine Learning", McGraw Hill.
2. C.M. Bishop, "Pattern Recognition and Machine Learning", Springer
3. Ethem Alpaydin, "Introduction to Machine Learning"

List of Assignments

1. Introduction to Learning Systems (Structure, Goals, Need, Applications, Examples).
2. Explain in detail steps to choose Machine Learning Algorithm. Differentiate between different Machine Learning Models.
3. Study and implement Decision Tree using R/Python Programming.
4. Study and implement Support Vector Machines using R/Python Programming.
5. What is Regression? Implement Linear Regression using R/Python Programming.
6. Examine Classification and Regression. What are the issues regarding classification and regression.
7. WAP to Implement Handwritten Digit Classification.
8. Study and implement K-Means clustering algorithm.
9. Study and implement Neural Network to Read Handwriting.
10. Case study on SCIKIT-LEARN, WEKA tool for machine learning.

Project Based Learning :

1. Movie Recommendations with Movie lens Dataset.
2. Sales Forecasting with Walmart.
3. Stock Price Predictions.
4. Human Activity Recognition with Smartphones.
5. Wine Quality Predictions.
6. Breast Cancer Prediction.
7. Iris Classification.

8. Movie Recommendations with Movie lens Dataset

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Data Storage Networking

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	04 Hours/Week	University Examination:	60 Marks		
		Internal Assessment:	40 Marks	Lecture	04
		Total	100 Marks	Total	04

Course Objective:
Student should be able to understand the different data storage technologies and able to design data storage solutions for an organization.

Prerequisite: Computer Network

Course Outcomes: On completion of the course, students will have the ability to:
1. Understand the design of a data centre and storage requirements.
2. Use the various types of storage and their properties.
3. Describe physical and virtualization of storage.
4. Understand the backup, archiving regarding recovery and business continuity.
5. Describe the backup/ recovery topologies.
6. Describe local replication and remote replication technologies.

Unit I **08 Hours**
Introduction to Storage Technology: Information storage, evolution ofstorage technology and architecture, data centre infrastructure, key challenges in Managing information, information lifecycle. Storage system Environments: components of storage system environment, Disk Drive components, Disk Drive Performance, fundamental laws governing disk performance, logical components of the host, application requirements and disk performance.

Unit II **08 Hours**
Data Protection: RAID: Implementation of RAID, RAID array components, RAID levels, RAID comparison, RAID Impact on disk performance, host spares. Intelligent Storage System: Components of an Intelligent Storage System, Intelligent Storage array, concepts in Practice: EMC CLARIION and Symmetric.

Unit III **08 Hours**
Direct – Attached Storage and Introduction to SCSI :Types of DAS,DAS benefits and limitations, disk drive interfaces, introduction to parallel SCSI, SCSI command model. Storage Area Networks: fibre channel, The SAN and Its evolution, components of SAN, FC connectivity, Fibre channel ports, fibre channel architecture, zoning, fiber channel login types, concepts in practice: EMC Connectix.

Unit IV **08 Hours**
Network attached storage: general purpose servers vs NAS Devices, benefits of NAS, NAS file I/O, components of NAS, NAS Implementations, NAS file sharing protocols, NAS I/O operations, factors effecting NAS Performance and availability, concepts in practice: EMC Celerra. IP SAN: iscsi, fcip. Content – addressed storage: Fixed content and Archives, types of archives, features and benefits of CAS, CAS Architecture, object storage and retrieval in CAS, CAS Examples, concepts in practice: EMC Centera.

08 Hours

Unit V

Storage Virtualization: Formas of Virtualization, SNIA Storage virtualization taxonomy, storage virtualization configurations, storage virtualization challenges, types of storage virtualization, concepts in practice: EMC In vista, Rainifinity. Introduction to business continuity: information availability, BC terminology, BC planning life cycle, Failure analysis, business impact analysis, BC technology solutions

Unit VI **08 Hours**

concepts in practice: EMC Power path. Backup and recovery: backup purpose, backup considerations, backup granularity, recovery considerations, backup methods, backup process, backup and restore operations, backup topologies, backup in NAS environments, backup technologies, concepts in practice: EMC Networker, EMC Disk Library(EDL).

Textbooks

- Mauricio Arregoces, Data Center Fundamentals, Cisco Press; 1st edition, 2003.
- Robert Spalding, Storage Networks: The Complete Reference, Tata McGraw Hill, Osborne, 2003
- Meeta Gupta, Storage Area Network Fundamentals, Pearson Education Limited, 2002.

Reference Books

- G. Somasundaram, Alok Shrivastava, Information Storage and Management, EMC Education Series, Wiley, Publishing Inc., 2011.
- Gustavo Santana, Data Center Virtualization Fundamentals: Understanding Techniques and Designs for Highly Efficient Data Centres with Cisco Nexus, UCS, MDS, and Beyond, Cisco Press; 1 edition, 2013.

Project Based Learning:

1. Implement different storage technology (File, Block and Object storage).
2. Implement Configuration of RAID on your Computer.
3. Create and Implement Google cloud console account and projects.
4. Study about implementation of EMC CLARIION and Symmetric.
5. Comparing SAN with standalone storages in server
6. Modeling /simulation of FC SAN -- Fibre Channel Storage Area Network
7. Implementation of FCoE – Fibre Channel over Ethernet.
8. Work flow management in cloud storage network.
9. How to model your project with assessments of storage and I/O workload requirements.
10. Cloud federation and also in green data centers.
11. Credential and also trust management in storage networking
12. Dynamic resource (resource as Data Storage) provisioning

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Data Visualization

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
	Hours/Week		**Marks**		**Credits**
Lecture:	3 Hours/Week	**University Examination**	**60 Marks**	**Lecture:**	03
Practical:	2 Hours/Week			**Practical:**	01
		Internal Assessment	**40 Marks**		
		Term Work	**25 Marks**		
		Practical	**25 Marks**		
		Total	**150 Marks**		04

Course Objective: Students will learn: the value of visualization, specific techniques in information visualization and scientific visualization, and how understand how to best leverage visualization methods.

Prerequisite: Programming Language, DBMS, JavaScript and HTML5

Course Outcomes: On completion of the course, students will have the ability to:

1. Explore various data visualization techniques.
2. Apply appropriate data visualization techniques to provide trends/insights for the dataset.
3. Apply visualization tools / techniques for various data analysis tasks.
4. Apply visualization tools / techniques for large datasets.
5. Apply advanced data visualization tools and techniques.
6. Given the application context for given data set, Design the information Dashboard for access information based on user criteria.

Unit I: Introduction to Data Visualization: **06 Hours**

Acquiring and Visualizing Data, Simultaneous acquisition and visualization, Applications of Data Visualization, Keys factors of Data Visualization (Control of Presentation, Faster and Better JavaScript processing, Rise of HTML5, Lowering the implementation Bar) Exploring the Visual Data Spectrum: charting Primitives (Data Points, Line Charts, Bar Charts, Pie Charts, Area Charts), Exploring advanced Visualizations (Candlestick Charts, Bubble Charts, Surface Charts, Map Charts, Infographics). Making use of HTML5 CANVAS, Integrating SVG.

Unit II: Basics of Data Visualization – Tables: **06 Hours**

Reading Data from Standard text files (.txt, .csv, XML), Displaying JSON content Outputting Basic Table Data (Building a table, Using Semantic Table, Configuring the columns), Assuring Maximum readability (Styling your table, Increasing readability, Adding dynamic Highlighting), Including computations, Using data tables library, relating data table to a chart

Unit III: Visualizing data Programmatically: **06 Hours**

Creating HTML5 CANVAS Charts (HTML5 Canvas basics, Linear interpolations, A Simple Column Chart, Animations), Starting with Google charts (Google Charts API Basics, A Basic bar chart, A basic Pie chart, Working with Chart Animations).

Unit IV: Introduction to D3.js: **06 Hours**

Getting setup with D3, Making selections, changing selection's attribute, Loading and filtering External data: Building a graphic that uses all of the population distribution data,

Data formats you can use with D3, Creating a server to upload your data, D3's function for loading data, Dealing with
Asynchronous requests, Loading and formatting Large Data Sets

Unit V: Advanced Data Visualization: **06 Hours**

Making charts interactive and Animated: Data joins, updates and exits, interactive buttons, Updating charts, Adding transactions, using keys.

Unit VI: Information Dashboard Design: **06 Hours**

Introduction, Dashboard design issues and assessment of needs, Considerations for designing dashboard-visual perception, Achieving eloquence, Advantages of Graphics _Library of Graphs, Designing Bullet Graphs, Designing Sparklines, Dashboard Display Media, Critical Design Practices, Putting it all together - Unveiling the dashboard.

Textbooks:

1. Jon Raasch, Graham Murray, Vadim Ogievetsky, Joseph Lowery, "JavaScript and jQuery for Data Analysis and Visualization", WROX.
2. Ritchie S. King, Visual story telling with D3" Pearson
3. Ben Fry, "Visualizing data: Exploring and explaining data with the processing environment", O'Reilly, 2008.
4. A Julie Steele and Noah Iliinsky, Designing Data Visualizations: Representing Informational Relationships, O'Relly.

Reference Books:

1. Scott Murray, Interactive Data Visualization for Web, O'Relly
2. Nathan Yau, "Data Points: Visualization that means something", Wiley, 2013.

List of Assignments

14. Setup Environment for All the Tools
15. Develop the following Program Using HTML5 CANVAS and SVG TAG
    - a.Develop the Different basic Graphical Shapes using HTM5 CANVAS
    - b.Develop the Different Advanced Graphical Shapes using HTM5 CANVAS
    - c.Develop the Different basic Graphical Shapes using HTM5 SVG
    - d.Develop the Different Advanced Graphical Shapes using HTM5 SVG
16. Develop Following Program Using HTML5 and JavaScript
    - a.Develop the simple bar chart usingTML5 CANVAS
    - b.Read the data .txt file and draw Data Table
    - c.Read the data .txt file and draw Simple Bar Chart
    - d.Read the data .csv file and draw Data Table
17. Develop Following Program Using HTML5 and JavaScript
    - a.Read the data .csv file and draw Column Bar Chart
    - b.Read the data XML file and draw Data Table
    - c.Read the data XML file and draw Simple Chart
    - d.Read JSON Data and draw Data Table
    - e.Read JSON Data and draw Simple Chart
18. Develop Following Program Using HTML5 and D3.js and Canvas.js
    - a.Showing the data as a column chart (simple)
    - b.Showing the data as a stacked column chart
    - c.Showing the Data as a column chart for four age group
    - d.Showing the data as a Line chart (single, fewer and multiple lines)
    - e.Showing the data as a Pie Chart (single and multiple pie)
    - f.Showing the data as a Bar Chart (Simple and multiple)
19. Develop Following Program Using HTML5 and Google Chats API and Map API

a.Using Google Charts API Basics draw charts like a Bar chart
b.Using Google Charts API Basics draw charts like a Line chart

20. Develop Following Program Using HTML5 and Google Chats API and Map API
    a.Using Google Charts API Basics draw PieChart.
    b.Using Google Charts API Basics draw Donut Chart.
    c.Using Google Charts API Basics draw Candle Chart
21. Develop Following Program Using HTML5 and Google Chats API and Map API
    a.Using Google Charts API Basics draw other types of Chart.
    b.Using Google API read JSON file and create Google Map.
22. Development of Dashboard.
23. Case Study

Project Based Learning :

1. Scatter Plot with Matplotlib.
2. Horizontal Bar Chart using Pandas.
3. Boxplot with Seaborn.
4. Histogram with Plotnine (ggplot).
5. Stacked Bar Plot.
6. Heatmaps.
7. Interactive Plot with Plotly (using Cufflinks).
8. Basic Interactive Binned Scatter Plot with Altair.
9. Correlogram.
10. Interactive Time Series Visualization.
11. Interactive Sunburst Charts.
12. Race Bar Chart.
13. Interactive Choropleth Map.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Elective II: Intelligent Autonomous Systems & Robotics

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	04 Hours/Week	University Examination:	60 Marks	Lecture	04
Tutorial	01 Hours/Week	Internal Assessment:	40 Marks	Tutorial	01
		Total	100	Total	05

Course Objective: To provide students with a working knowledge of methods for design and analysis of robotic and intelligent autonomous systems.

Prerequisite: Artificial Intelligence.

Course Outcomes: On completion of the course, students will have the ability to:

13. Familiarise with anatomy of Robots
14. Obtain forward and inverse kinematic models of robotic manipulators
15. Classify Robot End effectors.
16. Classify different types of sensors
17. Plan trajectories in joint space & Cartesian space
18. Familiarise with different types of Robotics applications

Unit I Introduction **08 Hours**

Types of Robots, Robotic system and robot Anatomy specification of robots, Robot configurations-PPP, RPP, RRP, RRR; features of SCARA, PUMA Robots; Classification of robots based on motion control methods and drive technologies, Flexible automation

Unit II Robot kinematics and dynamics **08 Hours**

Forward and Reverse Kinematics, open kinematic vs closed kinematic chain; degrees of freedom, Kinematic equation using Homogeneous Transformation, inverse Kinematics, Robot arm Dynamics, The Denavit-Hartenberg (D-H) representation, The arm equation, direct kinematics problems (upto 3DOF), Mobile Robot Kinematics

Unit III Effectors **08 Hours**

Classification of End effectors - mechanical grippers, special tools, Magnetic grippers, Vacuum grippers, adhesive grippers, Active and passive grippers, hooks and scoops, selection and design considerations of grippers in robot. Gripper force analysis and gripper design,

Unit IV Sensors and intelligent robots **08 Hours**

AI and automated manufacturing, Sensing system, Types of sensors, Robot Vision system, Design and control of sensor integrated Robot hand, Sensors for Mobile Robots, Sensor classification, Characterizing sensor performance, Representing uncertainty, Wheel/motor sensors, Heading sensors, Accelerometers, Inertial measurement unit (IMU), Ground beacons, Active ranging, Motion/speed sensors, Vision sensors

Unit V Trajectory planning **08 Hours**

Path planning, Trajectory Planning. Joint space trajectory planning- cubic polynomial, linear trajectory with parabolic blends, trajectory planning with via points; Cartesian space planning, Point to point vs continuous path planning. Obstacle avoidance methods- Artificial Potential field, A* algorithms

Unit VI Applications **08 Hours**

Robot Pose Maintenance and Localization: Simple Landmark Measurement, Servo Control, Recursive Filtering, Global Localization. Mapping: Sensorial Maps, Topological Maps, Geometric Maps, Exploration. Robots in Practice: Delivery Robots, Intelligent Vehicles, Mining Automation, Space Robotics, Autonomous Aircrafts, Agriculture, Forestry, Domestic Robots.

Textbooks

1. Deepak Khemani, "A First Course in Artificial Intelligence", McGraw Hill Education(India), 2013, ISBN :978-1-25-902998-1
2. Robotics Technology and Flexible Automation, Second Edition, S. R. Deb
3. Introduction to Robotics (Mechanics and control), John. J. Craig, Pearson Education Asia 2002.
4. Michael Jenkin, Gregory, " Computational Principals of Mobile Robotics", Cambridge University Press, 2010, ISBN : 978-0-52- 187157-0

Reference Books

4. Sicilliano, Khatib , "Handbook of Robotics", Springer
5. Andries P.Engelbrecht-Computational Intelligence: An Introduction, 2nd Edition-Wiley India- ISBN: 978-0-470-51250-0

Project Based Learning:

1. Robotic Arm Controlled by Touch Screen Display
2. Metal Detector Robotic Vehicle
3. **Chess Playing Robot**
4. **Autonomous Underwater Robot**
5. Sensor Guided Robotics
6. Smart Umbrella with Solar Cell
7. Artificial Intelligence-Based Chatbot for Appliance Control
8. Autonomous Quadcopter Docking System
9. Solar Panel Cleaning System Using Arduino
10. Pick and Place Robotic Vehicle

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Elective II: Deep Learning

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
	Hours/Week		Marks		Credits
Lecture:	4 Hours/Week	University Examination	60 Marks	Lecture:	04
Tutorial:	1 Hours/Week			Tutorial:	01
		Internal Assessment	40 Marks		
		Total	100 Marks		05

Course Objective: Introduce major deep learning algorithms, the problem settings, and their applications to solve real world problems.

Prerequisite: Linear Algebra, Statistics, probability, Machine learning

Course Outcomes: On completion of the course, students will have the ability to:

1. To Understand a wide variety of learning algorithms
2. To study the concepts of deep learning
3. To enable the students to know deep learning techniques to support real-time applications
4. Identify the deep learning algorithms which are more appropriate for various types of learning tasks in various domains.
5. Design and implement various deep supervised learning architectures for text & image data and design and implement various deep learning models and architectures
6. Apply various deep learning techniques to design efficient algorithms for real-world applications

Unit I: Basics of Neural Networks: Neural Networks basics – Binary Classification, Logistic Regression, Gradient Descent, Derivatives, Computation graph, Vectorization, Vectorizing logistic regression – **Shallow neural networks:** Activation functions, non-linear activation functions, Backpropagation, Data classification with a hidden layer. **08 Hours**

Unit II: Deep Neural Networks: Deep L-layer neural network, Forward and Backward propagation, Deep representations, Parameters vs Hyperparameters, Effective training in Deep Net- early stopping, Dropout, Batch Normalization, Instance Normalization, Group Normalization, Building a Deep Neural Network (Application). **08 Hours**

Unit III: Supervised Learning with Neural Networks: Practical aspects of Deep Learning: Train/Dev / Test sets, Bias/variance, Overfitting and regularization, Linear models and optimization, Vanishing/exploding gradients, Gradient checking. **08 Hours**

Unit IV: Logistic Regression, Convolution Neural Networks, RNN and Backpropagation – Convolutions and Pooling – Optimization algorithms: Mini-batch gradient descent, exponentially weighted averages, RMSprop, learning rate decay, problem of local optima, Batch norm – Parameter tuning process. Unsupervised Learning with Deep Network, Autoencoders. **08 Hours**

Unit V: Neural Network Architectures: Recurrent Neural Networks, Adversarial NN, Spectral CNN, Self-Organizing Maps, Restricted Boltzmann Machines, Recent Trends in Deep Learning Architectures, Residual Network, Skip Connection Network, Fully Connected CNN etc. **08 Hours**

Unit VI Long Short-Term Memory Networks (LSTM) and Deep Reinforcement Learning: TensorFlow, Keras or MatConvNet for implementation. Generative Modeling with DL, Variational Autoencoder, Generative Adversarial Network Revisiting Gradient Descent, Momentum Optimizer, RMSProp, Adam **08 Hours**

Textbooks

1. Deep Learning (Adaptive Computation and Machine Learning series) Hardcover, by Aaron Courville, Ian Goodfellow, Yoshua Bengio
2. Deep Learning for Natural Language Processing: Applications of Deep Neural Networks to Machine Learning Tasks by Pearson Learn IT
3. Advanced Deep Learning with Keras by Rowel Atienza
4. Deep Learning with Python Paperback – 22 December 2017
5. Advanced Deep Learning with Keras by Rowel Atienza

Reference Books

1. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow by Aurélien Géron
2. Machine Intelligence: Demystifying Machine Learning, Neural Networks and Deep Learning by Suresh Samudrala

Project Based Learning:

1. KNN (K - nearest neighbor) method
2. Artificial Neural Network (ANN)
3. Convolutional Neural Network (CNN)
4. Recurrent Neural Network (RNN)
5. Deep Neural Network (DNN)
6. Deep Belief Network (DBN)
7. Back Propagation
8. Stochastic Gradient Descent

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Elective II: Blockchain & Cryptocurrency

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		Marks		Credits
Lecture:	04 Hours/Week	University Examination:	60 Marks	Lecture	04
Tutorial:	01 Hours/Week	Internal Assessment:	40 Marks	Tutorial	01
		Total	100 Marks	Total	05

Course Objective:
To get acquainted with the concept of Block and Blockchain.
To analyze the applications& case studies of Blockchain.

Prerequisite:
Cyber Security, Network security, Distributed networks, Object Oriented programming language.

Course Outcomes: On completion of the course, students will have the ability to:

7. Describe the basic concept of Block chain.
8. Associate knowledge of consensus and mining in Block chain.
9. Summarize the bit coin crypto currency at an abstract level.
10. Apply the concepts of keys, wallets and transactions in the Bit coin network.
11. Interpret the knowledge of Bit coin network, nodes and their roles.
12. Illustrate the applications of Block chain and analyze case studies.

Unit I Introduction to Block chain **08 Hours**

Structure of a Block, Block Header, Block Identifiers: Block Header Hash and Block Height, The Genesis Block, Linking Blocks in the Block chain, Merkle Trees and Simplified Payment Verification (SPV).

Unit II Consensus and Mining **08 Hours**

Decentralized Consensus, Byzantine General's Problem, Independent Verification of Transactions, Mining Nodes, Aggregating Transactions into Blocks, Constructing the Block header, Mining the Block, Successfully Mining the Block, Validating a New Block, Assembling and Selecting Chains of Blocks, Block chain Forks

Unit III Introduction to Bit coin **08 Hours**

What is Bit coin and the history of Bit coin, Getting the first bit coin, finding the current price of bit coin and sending and receiving bit coin, Bit coin Transactions.

Unit IV Concepts of Bit coin **08 Hours**

Keys and addresses, Wallets and Transactions: Public Key Cryptography and Crypto currency, Private and Public Keys, Bit coin Addresses, Base58 and Base58Check Encoding, Nondeterministic (Random) Wallets, Deterministic (Seeded) Wallets, HD Wallets (BIP-32/BIP-44), Wallet Best Practices, Using a Bit coin Wallets, Transaction Outputs and Inputs, Transaction Fees, Transaction Scripts and Script Language, Turing Incompleteness, Stateless Verification, Script Construction (Lock + Unlock), Pay-to-Public-Key-Hash (P2PKH), Bitcoin Addresses, Balances, and Other Abstractions

Unit V Bit coin Networks **08 Hours**

Peer-to-Peer Network Architecture, Node Types and Roles, Incentive based Engineering The Extended Bitcoin Network, Bitcoin Relay Networks, Network Discovery, Full Nodes, Exchanging "Inventory", Simplified Payment Verification (SPV) Nodes, Bloom Filters, SPV Nodes and Privacy, Encrypted and Authenticated Connections, Transaction Pools

Unit VI Blockchain Applications & case studies **08 Hours**

Domain-Specific Applications: FinTech, Internet of Things, Industrial and Manufacturing, Energy, Supply chain & Logistics, Records & Identities, Healthcare Case studies related to cryptocurrencies Concept of Altcoin.

Textbooks

6. Mastering Bitcoin, PROGRAMMING THE OPEN BLOCKCHAIN, 2nd Edition by Andreas M. Antonopoulos, June 2017, Publisher(s): O'Reilly Media, Inc. ISBN:9781491954386.
7. Blockchain Applications: A Hands-On Approach", by Arshdeep Bahga, Vijay Madisetti, Paperback – 31 January 2017

Reference Books

4. "Mastering Blockchain", by Imran Bashir, Third Edition, Packt Publishing
5. "Mastering Ethereum: Building Smart Contracts and Dapps Paperback" by Andreas Antonopoulos, Gavin Wood, Publisher(s): O'Reilly Media

Project Based Learning - Provisional List of Projects

1. Smart Contract: Development of smart block-based contact for project development
2. Crypto-wallet: Creating a Crypto wallet for handling cryptocurrency
3. Cryptocurrency: ERC-20 tokens & creating own cryptocurrency using solidity for Ethereum.
4. Blockchain-based Lottery – Picking a Winner from various Blockchain Nodes taking part in a lottery.
5. Install and Use Ganache, Flask and Postman.
6. Remix-Ethereum IDE
7. Simple Smart Contract for Bank with withdraw and deposit functionality.
8. Smart Contract for storing and retrieving information of Degree Certificates.
9. Simple Python program to create a Block class that contains index, timestamp, and previous hash. Connect the blocks to create a Blockchain

Syllabus for Unit Tests

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Elective II: Docker and Kubernetes

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
	Hours/Week		Marks		Credits
Lecture:	4 Hours/Week	University Examination	60 Marks	Theory	04
Tutorial:	1 Hours/Week			Tutorial	01
		Internal Assessment	40 Marks		
		Total	150 Marks	Total	05

Course Objective:
To make student aware of basics of Docker and Kubernetes Knowledge representation methods, learning concept and basics of it.

Prerequisite:
DevOps, Linux, AWS, and Docker concepts, CI + CD pipeline

Course Outcomes: On completion of the course, students will have the ability to:

1. Describe the challenges in developing Docker Syatem
2. Apply appropriate problem-solving strategy to solve a particular problem
3. Use appropriate knowledge representation method
4. Describe components of planning system
5. Apply the various knowledge representation strategies
6. Describe the plan generation systems

Unit I: Introduction to Docker **08 Hours**

What is Docker , manipulating container with docker client , Docker run in detail, Container lifecycle and Log output, Purpose of IT flag,
Starting with shell, Docker Architecture

Unit II: Building Custom Images through Docker Server **08 Hours**

Creating Docker image, Base Image , Build Process, Tagging in image .Node Server Setup , Base image issues , Copying build files, Container port mapping, Specifying a Working Directory , Unnecessary Rebuilds, Minimizing Cache Busting and Rebuilds.

Unit III Docker Architecture and CICD with AWS **08 Hours**

The Docker daemon, The Docker client, Docker registries

Docker objects, Development work flow, Docker volume , Shorthand with Docker Compose, Multi-Step Docker Builds, Github Setup, Travis CI Setup, Travis YML File Configuration, AWS Elastic Beanstalk, Travis Config for Deployment, Automated Deployments, Exposing Ports Through the Dockerfile, Workflow With Github, Redeploy on Pull Request Merge, Single Container Deployment Issues, Checkpoint Catchup, Adding Postgres as a Service, Docker-compose Config, Production Dockerfiles, Creating the EB Environment, Managed Data Service Providers, Verifying Deployment

Unit IV Kubernetes **08 Hours**

Overview , Features , Kubernetes - Cluster Architecture, Kubernetes - Master Machine Components, etcd, API Server, Controller Manager, Scheduler, Kubernetes - Node Components, Docker, Kubelet Service, Kubernetes Proxy Service, Kubernetes - Master and Node Structure

Unit V: Kubernetes services **08 Hours**

Service without Selector, Service Config File with Selector , Multi-Port Service Creation, Types of Services ClusterIP , NodePort, Load Balancer , Types of Pod , Kubernetes - Replication Controller, Kubernetes - Replica Sets, Kubernetes – Volumes , Kubernetes – Secrets , Kubernetes - Network Policy

Unit VI : Advanced Kubernet **08 Hours**

API, Kubectl , Kubectl Commands, Creating an App, Autoscaling, Dashboard Setup, Monitoring

Textbooks

1. The Docker Book James Turnbull in 2014
2. Docker up and running Karl Matthias and Sean P

Reference Books

Dr. Gabriel N. Schenker, “Containerize your Apps with Docker and Kubernetes”.

Project Based Learning - Provisional List of Projects

1. Creating a docker image
2. Running the first container
3. Retrieving container logs
4. Creating CI/CD pipeline
5. Writing script for build automation
6. With the help of App infrastructure isolation (Deploying multiple apps on docker)
7. Use Multi-tenancy support(working with different computing environments)
8. Creation of microservices architecture
9. Installing docker on VM
10. Build and deploy the multistage image

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Programming Technologies and Tools Laboratory – VII

Teaching Scheme		Examination Scheme		Credit Scheme	
	Hours/Week		**Marks**		**Credits**
Lecture:	**01 Hours/Week**	**Term Work**	**25 Marks**		
Practical:	**02 Hours/Week**	**Practical**	**25 Marks**	**Lecture**	**01**
			50 Marks	**Practical**	**01**
		Total	**50 Marks**	**Total**	**02**

Course Objective:

3. Understand role of blockchain in Web 3.0.
4. Understand the bitcoin blockchain platform and its terminologies.
5. Understand Ethereum architecture and enterprise blockchain.
6. Study and developed smart contracts, DAPPS for different application.

Prerequisite:

1. Cyber Security, Network security, Distributed networks
2. Object Oriented programming language.

Course Outcomes:

1. Differentiate between Web 2.0 and Web 3.0 with respect to various applications.
2. Elaborate the bitcoin mining, DLT, Consensus algorithm.
3. Analyse the Ethereum architecture.
4. Analyse the Hyperledger Fabric architecture.
5. Design smart contract and DAPP for real time application.
6. Illustrate blockchain integration with emerging technologies and security issues.

Unit I: Fundamentals of Blockchain **02 Hours**

Challenges Faced by Modern Businesses, Features of Blockchain, Building Blocks of Blockchain, Introduction to Blockchain Pillars, Why Blockchain Platform: Platform types, Public, Private, technology requirements for implementation. Distributed Ledger, Introduction to cryptography-Encryption and Decryption-Ciphers-Cryptography using arithmetic modulo primes-hashing algorithms-SHA-256 algorithm-Application of SHA algorithm, Web 2.0 and Web 3.0.

Unit II: Bitcoin Blockchain **02 Hours**

Introduction to Bitcoin, Bitcoin Wallets, Bitcoin Block, Bitcoin Transaction, Bitcoin Network, Operation of Bitcoin Blockchain, Blockchain Architecture – Block, Hash, Distributer P2P, Structure of Blockchain- Consensus mechanism: Proof of Work, Bitcoin (BTC) – Genesis Block, Buy Bitcoin, Transactions, Unspent Transaction Output (UTXO), Bitcoin Mining, Value of Bitcoin, Advantages and Disadvantages

Unit III: Ethereum Blockchain **02 Hours**

Introduction, Ethereum components: miner and mining node, Ethereum virtual machine, Ether, Gas, Transactions, accounts, swarm and whisper, Ethash, end to end transaction in Ethereum, architecture of Ethereum Dapp Architecture, DAO.

Unit IV: Ethereum Smart Contracts **02 Hours**

Smart Contract, Smart Contract Lifecycle, Solidity, Solidity State and Variable Types, Solidity Functions, Solidity Compilation and Deployment, mapper function, ERC20 and ERC721 Tokens, comparison between ERC20 & ERC721, ICO, use cases of smart, contract, smart Contracts: Opportunities, Risks.

Unit V Enterprise Blockchain **02 Hours**

Introduction to Hyperledger, tools and frameworks, Hyperledger Fabric, Comparison between Hyperledger Fabric & Other Technologies, Distributed Ledgers. Hyperledger Fabric Architecture, Components of Hyperledger Fabric: MSP, Chain Codes etc., Transaction Flow, Advantages of Hyperledger Fabric Blockchain, working of Hyperledger Fabric, Creating Hyperledger network, Case Study of Supply chain.
management using Hyperledger. Hyperledger Fabric (A): Decomposing the consensus process, Hyperledger fabric components, Chain code Design and Implementation Hyperledger Fabric (B): Beyond Chain code: fabric SDK and Front End (b) Hyperledger composer tool.

Unit VI: Blockchain integration and Research challenges **02 Hours**

Integrating Blockchain with cloud, IoT, AI, ERP, End to end blockchain integration, Risks and Limitations of Blockchain: Privacy & Security. Criminal Use of Payment Blockchains, The "Dark" Side of Blockchain.

Textbooks

9. "Mastering Bitcoin, PROGRAMMING THE OPEN BLOCKCHAIN" , 2nd Edition by Andreas M. Antonopoulos

10. William Stallings, "Cryptography and Network security Principles and Practices", Pearson/PHI,2017

11. Mastering Ethereum, Building Smart Contract and Dapps, Andreas M. Antonopoulos Dr. Gavin Wood, O'reilly.

Reference Books

1. Mastering Blockchain", by Imran Bashir, Third Edition,Packt Publishing
2. Blockchain with Hyperledger Fabric,LucDesrosiers, Nitin Gaur, Salman A. Baset, Venkatraman Ramakrishna,Packt Publishing
3. Atul Kahate, Cryptography and Network Security, Tata Mc Grawhill, India, 2019.

List of Experiments:

1. Working of Blockchain Transaction, DLT(https://andersbrownworth.com/blockchain/)
2. Implement program to convert given text in to hashes using SHA 256 algorithm.
3. Create simple wallet transaction from one account to another account using Metamask.
4. Connect Metamask to a Ganache Test Network
5. Ether Transaction Using Ganache.
6. Write Hello World smart contract in a higher programming language (Solidity).

7. Write simple smart contract for User identity management using Solidity language.
8. Write simple smart contract for Crowd fund ERC20 token

    1. User creates a campaign.
    2. Users can pledge, transferring their token to a campaign.
    3. After the campaign ends, campaign creator can claim the funds if total amount pledged is more than the campaign goal.
    4. Otherwise, campaign did not reach its goal, users can withdraw their pledge.

9. Build NFT Application for Fan engagement and gaming rewards
10. Write smart contract for Tracking property details in real estate
    Create DAPP for Protecting sensitive medical data in healthcare
11. Build NFT Application by writing smart contract
    English Auction
    English auction for NFT.
    Auction
    - Seller of NFT deploys this contract.
    - Auction lasts for 7 days.
    - Participants can bid by depositing ETH greater than the current highest bidder.
    - All bidders can withdraw their bid if it is not the current highest bid.

    After the auction
    - Highest bidder becomes the new owner of NFT.
    - The seller receives the highest bid of ETH.
12. Creating a Business Network using Hyperledger Fabric

Syllabus for Unit Tests: NA

Project Stage -II

TEACHING SCHEME	**EXAMINATION SCHEME**	**CREDIT SCHEME**	
			Credits
Practical: 4 Hrs/Week		Practical	06
	Term Work : 100 Marks	Term Work	
	Oral : 100 Marks	Oral	
		Term Work	
	Total 200 marks	Total	06

Course Pre-requisites:
Basics of Software engineering, Software testing and knowledge of core computer engineering subjects.

Course Objectives:

- To provide in depth outline for software project planning and development.

Course Outcomes: On completion of the course, students will have the ability to:

- Use appropriate software development tool for the proposed problem.
- Deign test cases and perform testing.
- Perform collaboratively towards a common purpose.
- Demonstrate self-advocacy skills and self-reliant behaviour.
- Demonstrate the ability to develop and maintain satisfying interpersonal. relationships.
- Evaluate and conclude the results with documentation.

1 The project will be undertaken preferably by a group of at least 3- 4 students who will jointly work and implement the project over the academic year. The work will involve the design of a system or subsystem in the area of Computer Engineering.

2. If the project is chosen a hardware project it will involve the designing a system – subsystem or upgrading an existing system. The design must be implemented into a working model with necessary software interfacing and a user manual.

3. If the project is chosen in the pure Software Application it must involve the detail Software Design Specifications, Data Structure Layout, File Design, Testing with complete documentation and user interface. With life cycle testing and as an executable package.

The group will submit at the end of Semester-VIII,

i) The workable project.

ii) The details of Research paper published in National/International paper conferences/journals for the project work carried out.

iii) Project Report complete in all aspects, 3 copies for the institute and 1 copy of each student in the group for certification.

The examiner in consultation with the guide will assess the term work.

Oral examination will be based on the project work completed by the candidate.

Project report must be checked for plagiarism from respective guide.

The project report will contain the following details:

1. Problem definition and requirement specification, acceptance tests procedure (ATP).
2. System definition, requirement analysis.
3. System design.
4. System implementation-code documentation –dataflow diagram / algorithm.
5. Test results and procedure, test report as per ATP.
6. Platform choice, use.
7. Appendix tools used, references.
8. Documentation will use UML approach with Presentation, Category, Use Case, Class Diagrams, etc.

BHARATI VIDYAPEETH
(DEEMED TO BE UNIVERSITY), PUNE

Faculty of Engineering & Technology
B.Tech (Computer Engineering)
Old Syllabus

BHARATI VIDYAPEETH
(DEEMED TO BE UNIVERSITY) Pune.

Faculty of Engineering & Technology
B.Tech. (Computer Science and Engineering)
Program Curriculum
(2014 Course)

COURSE STRUCTURE AND SYLLABUS
(Choice Based Credit System - 2014 Course)
B.Tech. (Computer Engineering)
Program Curriculum Sem I & II

Programme: B. Tech.Computer Engineering

VISION OF UNIVERSITY:

To be World Class Institute for Social Transformation Through Dynamic Education.

MISSION OF UNIVERSITY:

A. To provide quality technical education with advanced equipment, qualified faculty members, infrastructure to meet needs of profession and society.

B. To provide an environment conducive to innovation, creativity, research, and entrepreneurial leadership.

C. To practice and promote professional ethics, transparency and accountability for social community, economic and environmental conditions.

VISION OF THE DEPARTMENT

To pursue and excel in the Endeavour for creating globally recognized Computer Engineers through Quality education.

MISSION OF THE INSTITUTE:

- To provide quality technical education with advanced equipment, qualified faculty members, infrastructure to meet needs of profession and society.
- To provide an environment conducive to innovation, creativity, research, and entrepreneurial leadership.
- To practice and promote professional ethics, transparency and accountability for social community, economic and environmental conditions.

MISSION OF THE DEPARTMENT

A. To impart engineering knowledge and skills confirming to a dynamic curriculum.

B. To develop professional, entrepreneurial & research competencies encompassing continuous intellectual growth.

C. To produce qualified graduates exhibiting societal and ethical responsibilities in working environment.

PROGRAM EDUCATIONAL OBJECTIVES

1. Demonstrate technical and professional competencies by applying engineering fundamentals, computing principles and technologies.
2. Learn, Practice, and grow as skilled professionals/ entrepreneur/researchers adapting to the evolving computing landscape.

3. Demonstrate professional attitude, ethics, understanding of social context and interpersonal skills leading to a successful career.

PROGRAM SPECIFIC OUTCOMES

PSO 1. Apply fundamental knowledge and technical skills towards solving Engineering problems.

PSO 2. Employ expertise and ethical practise through continuing intellectual growth and adapting to the working environment.

PROGRAM OUTCOMES

1. To apply the knowledge of computing and mathematics appropriate to domain.
2. To logically define analyze and solve real world problems.
3. To apply design principles in developing Hardware and software systems.
4. To interpret and analyze data for providing solutions.
5. To use and practice engineering and IT tools for professional growth.
6. To understand and respond to legal and ethical issues.
7. To develop social relevant projects using available resources.
8. To exhibits professional and ethical responsibilities.
9. To work effectively as an individual and a team member.
10. To prepare and present technical documents using effective communication skill.
11. To demonstrate effective leadership skills through project management life cycle.
12. To understand the significance of lifelong learning for professional development.

Bharati Vidyapeeth University, Pune

Faculty of Engineering & Technology

Programme: B. Tech (Computer Engineering) Sem – I & II (2014 Course)

Graduate Attributes/ Programme Outcomes	a	b	c	d	e	f	g	h	i	j	k	l
Engineering Knowledge	✓											
Problem Analysis		✓										
Design/Development of solutions			✓									
Conduct Investigations of Complex Problems				✓								
Modern Tool Usage					✓							
The Engineer and Society						✓						
Environment and Sustainability							✓					
Ethics								✓				
Individual and teamwork									✓			
Communication										✓		
Project management and finance											✓	
Life-long learning												✓

Elective - I: a) Multimedia and Mobile Applications b) Scripting Languages c) Software Project Management d) Computational Genomics

B. Tech. (Computer) - 2014
Course Semester –I

Sr. No	Subject	Teaching Scheme			Examination Scheme-Marks						Credits	
					End Semester Examination	Continuous Assessment			TW	Total		
		L	P/D	T		Unit Test	Attendance	Assignments			Th	TW
1	Engineering Mathematics - I	3	--	1	60	20	10	10	--	100	4	--
2	Fundamentals of Civil Engineering	3	2	--	60	20	10	10	25	125	3	1
3	Engineering Graphics	4	2	--	60	20	10	10	25	125	4	1
4	Engineering Chemistry	4	2	--	60	20	10	10	25	125	4	1
5	Fundamentals of Electrical Engineering	3	2	--	60	20	10	10	25	125	3	1
6	Professional Skill Development - I	2	--	--	30	--	--	20	--	50	2	--
7	Programming Principles and Paradigms	--	2	--	--	--	--	--	50	50	--	1
	Total	**19**	**10**	**1**	**330**	**100**	**50**	**70**	**150**	**700**	**20**	**5**

Sr. No	Subject	Teaching Scheme			Examination Scheme-Marks						Credits	
					End Semester Examination	Continuous Assessment			TW	Total		
		L	P/D	T		Unit Test	Attendance	Assignments			Th	TW
1	Engineering Mathematics - I	3	--	1	60	20	10	10	--	100	4	--
2	Fundamentals of Civil Engineering	3	2	--	60	20	10	10	25	125	3	1
3	Engineering Graphics	4	2	--	60	20	10	10	25	125	4	1
4	Engineering Chemistry	4	2	--	60	20	10	10	25	125	4	1
5	Fundamentals of Electrical Engineering	3	2	--	60	20	10	10	25	125	3	1
6	Professional Skill Development - I	2	--	--	30	--	--	20	--	50	2	--
7	Programming Principles and Paradigms	--	2	--	--	--	--	--	50	50	--	1
	Total	**19**	**10**	**1**	**330**	**100**	**50**	**70**	**150**	**700**	**20**	**5**

Teaching Scheme			Examination Scheme-Marks						Credits	
Lectures	Practical	Tutorial	End Semester Examination	Unit Test	Attendance	Assignments	TW	Total	Theory	TW
19	**10**	**1**	**330**	**100**	**50**	**70**	**150**	**700**	**20**	**5**

B. Tech. (Computer) - 2014 Course
Semester –II

Sr. No	Subject	Teaching Scheme			Examination Scheme-Marks						Credits	
					End Semester Exam.	Continuous Assessment			TW	Total		
		L	P/D	T		Unit Test	Attendance	Assignments			Th	TW
8	Engineering Mathematics-II	3	--	1	60	20	10	10	--	100	4	--
9	Fundamentals of Mechanical Engineering	3	2	--	60	20	10	10	25	125	3	1
10	Engineering Mechanics	4	2	--	60	20	10	10	25	125	4	1
11	Engineering Physics	4	2	--	60	20	10	10	25	125	4	1
12	Object Oriented Programming	3	2	--	60	20	10	10	25	125	3	1
13	Professional Skill Development - II	2	--	--	30	--	--	20	--	50	2	--
14	Workshop Technology	--	2	--	--	--	--	--	50	50	--	1
	Total	**19**	**10**	**1**	**330**	**100**	**50**	**70**	**150**	**700**	**20**	**5**

Teaching Scheme			Examination Scheme-Marks						Credits	
Lectures	Practical	Tutorial	End Semester Examination	Unit Test	Attendance	Assignments	TW	Total	Theory	TW
19	**10**	**1**	**330**	**100**	**50**	**70**	**150**	**700**	**20**	**5**

Total Marks of Semester –I and Semester-II = 1400

Total Credits of Semester –I and Semester-II = 50

ENGINEERING MATHEMATICS-I

Teaching Scheme	Examination Scheme	Credit Scheme
Lectures: 3Hrs/Week	Semester Examination: 60 marks	Theory : 03
Tutorials: 1Hr/Week	Continuous Assessment: 40 marks	Tutorial : 01

Unit I **08 Hours**

MATRICES

Rank, Normal form, System of Linear Equations, Linear Dependence and Independence, Linear and Orthogonal Transformations. Eigen values, Eigen Vectors, Cayley – Hamilton Theorem.Application to problems in Engineering .

Unit II **08 Hours**

COMPLEX NUMBERS AND APPLICATIONS

Definition, Cartesian, Polar and Exponential Forms ,Argand's Diagram, De'Moivre's theorem and its application to find roots of algebraic equations., Hyperbolic Functions, Logarithm of Complex Numbers, Separation into Real and Imaginary parts, Application to problems in Engineering.

Unit III **08 Hours**

DIFFERENTIAL CALCULUS

Successive Differentiation, nth Derivatives of Standard Functions,Leibnitz's Theorem.

EXPANSION OF FUNCTIONS

Taylor's Series and Maclaurin's Series.

Unit IV **08 Hours**

DIFFERENTIAL CALCULUS

Indeterminate Forms, L' Hospital's Rule, Evaluation of Limits.

INFINITE SERIES

Infinite Sequences, Infinite Series, Alternating Series, Tests for Convergence, Absolute and Conditional Convergence, Power series, Range of Convergence.

9

Unit V **08 Hours**

PARTIAL DIFFERENTIATION AND APPLICATIONS

Partial Derivatives, Euler's Theorem on Homogeneous Functions, Implicit functions, Total Derivatives, Change of Independent Variables. Errors and Approximations

Unit VI **06 Hours**

JACOBIAN

Jacobians and their applications, Chain Rule, Functional Dependence.

MAXIMA AND MINIMA

Maxima and Minima of Functions of two variables, Lagrange's method of undetermined multipliers.

Assignment:

1. Rank ,System of Linear Equations.
2. Complex Numbers.
3. Differential Calculus and Expansion of Functions.
4. Indeterminate Forms and Infinite Series.
5. Partial Derivatives, Euler's Theorem on Homogeneous Functions.
6. Jacobians, Maxima and Minima of Functions of two variables.

Textbooks:

1. Applied Mathematics (Volumes I and II) by P. N. Wartikar& J. N. Wartikar, Pune VidyarthiGrihaPrakashan, Pune, 7th edition (1988).
2. Higher Engineering Mathematics by B. S. Grewal, Khanna Publication, Delhi, 42th edition (2012).
3. Higher Engineering Mathematics by B.V. Ramana, Tata McGraw-Hill (2008) .
4. Advanced Engineering Mathematics by Erwin Kreyszig, Wiley Eastern Ltd, 8thedition(1999).
5. Advanced Engineering Mathematics, 7e, by Peter V. O'Neil, Thomson Learning,6th edition (2007).
6. Advanced Engineering Mathematics, 2e, by M. D. Greenberg, Pearson Education, 2nd edition(2002).

Syllabus for Unit Test:

Unit Test I :-Unit I,II,III

Unit Test II :-Unit IV,V,VI

02: Fundamentals of Civil Engineering

Teaching Scheme	Examination Scheme	Credit Scheme
Theory: 03 Hours / Week	End Semester Examination: 60 Marks	03 Credits
Practical: 02 Hours / Week	Continuous Assessment: 40 Marks	
	Term Work: 25 Marks	01 Credit

Course Pre-requisites:

The Students should have

1. Concepts of units and conversions of units.
2. Basic knowledge of Chemistry
3. Basic knowledge of geography, concept of latitude and longitude.

Course Objectives:

To make student understand the scope and application of Civil Engineering

Course Outcomes: Students will be able to understand

1. Different building components and material
2. Classification of surveying
3. Levellingof the ground
4. Planning of building
5. Methods of irrigation and water supply
6. Different methods of transportation

Unit I **06 Hours**

Civil Engineering Scope And Applications.

Civil Engineering scope, importance and applications to other disciplines of Engineering; Civil Engineering construction process and role of Civil engineer; Government authorities related to Civil Engineering; Types of structures based on loading , material and configuration; Building components and their functions; Civil Engineering materials: concrete, construction steel, bricks, flooring material and tiles, paints, plywood , glass and aluminum.

Unit II **06 Hours**

Surveying

Objectives, Principles and Classification of Surveying; Linear, angular, Vertical and area Measurements and related instruments.

Unit III **06 Hours**

Building Planning And Bye Laws

Site selection for residential building; Principles of building planning; Building bye laws- necessity, Floor Space Index, Heights , open space requirements, set back distance , ventilation and lighting, concept of carpet and built up area, minimum areas and sizes for residential buildings ; Concept of Eco friendly structures and Intelligent buildings.

Unit IV **06 Hours**

Foundations and Earthquakes

Function of foundation, concept of bearing capacity and its estimation, types of foundation and its suitability, causes of failure of foundation. Earthquakes causes, effects and guidelines for earthquake resistant design, earthquake zones.

Unit V **06 Hours**

Irrigation And Water Supply

Rainfall measurement and its use in design of dams; Types of dams, canals, methods of irrigation and their merits and demerits; hydropower structures ;Water supply, drinking water requirements and its quality, water and sewage treatment flow chart.

Unit VI **06 Hours**

Infrastructure

Roads- types of roads and their suitability, cross section of roads, meaning of terms ; width of roads, super elevation, camber, gradient ,sight distance, materials used for construction of roads.

Railways- Types of gauges, section of railway track, components of railway track, advantages.

Bridges: Components - Foundation, Piers, Bearings, Deck. Airways- Components -Runway, Taxiway and Hangers.

Term Work:

1. Study and use of prismatic compass and measurement of bearings.
2. Study and use of Dumpy level and reduction of levels by collimation plane method.
3. Area measurement by Digital Planimeter.
4. Drawing plan and elevation of a residential bungalow.
5. Study of features of topographical maps.
6. Assignment on collection of information on Civil Engineering materials.
7. Assignment on types of foundations.
8. Assignment problem on irrigation and hydropower structures.
9. Assignment on study of flow chart of water and sewage treatment.
10. Assignments on types of transportation systems.

Textbooks:

1. " Surveying- Vol I " - S.K. Duggal , Tata McGraw Hill Publication.
2. "Built Environment" – Shah , Kale, Patki, , Tata McGraw Hill Publication
3. "Building Construction" – Dr. B.C. Punmia , Laxmi Publication
4. "Irrigation and water Power Engineering "- Dr. P.N. Modi,Standard Publishers ,New Delhi
5. "Text book of Transportation Engineering "- Arora, Charotar Publishers.
6. Water supply and sanitary engineering-Rangawala, Charotar Publishers.
7. "Basic Civil engineering"- M.S. Palanichamy- Tata McGraw Hill Publication

Reference Books:

1. "Surveying –Theory and Practice"-James Anderson- Tata McGraw Hill Publication

Syllabus for Unit Tests:

Unit Test -1 Unit I to III

Unit Test -2 Unit IV to VI

ENGINEERING GRAPHICS

Teaching Scheme	Examination Scheme	Credit Scheme
Theory: -04 Hours / Week	End Semester Examination: -60Marks	
Practical: 02 Hours / Week	Continuous Assessment: -40Marks	
	Term Work: 25 Marks	

Unit I **06 Hours**

Lines and Dimensioning in Engineering Drawing

Different types of lines used in drawing practice, Dimensioning – linear, angular, aligned system, unidirectional system, parallel dimensioning, chain dimensioning, location dimension and size dimension.

Curves used in Engineering Practice

Ellipse by Directrix-Focus method, Arcs of Circle method, Concentric circle method and Oblong method. Involute of a circle, Cycloid, Archimedean Spiral, Helix on cone, Loci of points- Slider Crank mechanisms.

Unit II **06 Hours**

Orthographic Projection

Basic principles of orthographic projection (First and Third angle method). Orthographic projection of objects by first angle projection method only. Procedure for preparing scaled drawing, sectional views and types of cutting planes and their representation, hatching of sections.

Unit III **06 Hours**

Isometric Projections

Isometric view, Isometric scale to draw Isometric projection, Non-Isometric lines, and construction of Isometric view from given orthographic views and to construct Isometric view of a Pyramid, Cone, and Sphere.

Unit IV **06 Hours**

Projections of Points and Lines and planes

Projections of points, projections of lines, lines inclined to one reference plane, Lines inclined to both reference planes. (Lines in First Quadrant Only) Traces of lines, Projections of Planes, Angle between two planes, Distance of a point from a given plane, Inclination of the plane with HP, VP

Unit V **06 Hours**

Projection of Solids

Projection of prism, pyramid, cone and cylinder by rotation method.

Unit VI **06 Hours**

Section of Solids

Types of section planes, projections of solids cut by different sections of prism, pyramid, cone and cylinder.

Term Work:

Term work shall consist of five half-imperial size or A2 size (594 mm x 420 mm) sheets. Assignment 05 Problems on each unit in A3 size Drawing Book

SHEETS

1. Types of lines, Dimensioning practice, Free hand lettering, 1nd and 3rd angle methods symbol.
2. Curves and loci of points
3. Projections of Points and Lines and planes
4. Orthographic Projections
5. Isometric views
6. Projection of Solids

Text Books

1. "Elementary Engineering Drawing", N.D. Bhatt, Charotar Publishing house, Anand India,
2. "Text Book on Engineering Drawing", K.L.Narayana & P.Kannaiah, Scitech Publications, Chennai. 3."Fundamentals of Engineering Drawing", Warren J. Luzzader, Prentice Hall of India, New Delhi ,
4. "Engineering Drawing and Graphics", Venugopal K., New Age International Publishers.
5. M. B. Shah and B. C. Rana, "Engineering Drawing", 1st Ed, Pearson Education, 2005

. P. S. Gill, "Engineering Drawing (Geometrical Drawing)", 10 Edition, S. K. Kataria and Sons, 2005

7. P. J. Shah, "Engineering Drawing", C. Jamnadas and Co., 1 Edition, 1988

ENGINEERING CHEMISTRY

Teaching Scheme	Examination Scheme	Credit Scheme
Lectures: 4Hrs/Week	End Semester Examination: 60 marks	Theory: 04
Practical: 2Hr/Week	Continuous Assessment: 40 marks	Practical: 01
	Term Work: 25marks	

Unit I WATER (08 Hours)

Introduction, Hardness of water, Effect of hard water on boilers and heat exchangers: a) boiler corrosion b) caustic embrittlement c) scales and sludges d) priming and foaming Water softening methods for industrial purposes :a) Zeolite process b) Phosphate conditioning Numerical based on the zeolite process

Unit II 08 Hours

MATERIAL CHEMISTRY

Crystallography : Unit cell, Laws of crystallography, Weiss indices and Miller indices, Crystal defects (point and line defects), X-ray diffraction – Bragg's Law and numerical.

Cement : Introduction of cement, Hydraulic/ Non-hydraulic cementing materials, classification of cement, chemistry of portland cement, chemical composition and compound constituents of portland cement, properties of cement and its applications.

Unit III 08 Hours

FUELS

Introduction, classification of fuels, calorific value of fuels, NCV and GCV, Determination of calorific values using Bomb calorimeter and Boys' gas calorimeter.

Theoretical calculation of calorific value of a fuel, Analysis of coal a) Proximate b) Ultimate analysis of coal, Numericals based on NCV, GCV.

Unit IV **08 Hours**

CORROSION AND ITS PREVENTION

Corrosion: - Definition, atmospheric corrosion-mechanism, Wet corrosion-mechanism, Electrochemical and galvanic series, Factors affecting corrosion-nature of metal, nature of environment.

Methods of prevention of corrosion- Cathodic and Anodic protection, Metallic coatings, Electroplating, Hot dipping.

Unit V **08 Hours**

ELECTROCHEMISTRY

Introduction, Arrhenius Ionic theory, Kohlrausch's law of independent migration of ions Laws of electrolysis: Faradays Laws, Ostwald's dilution law, Acids and Bases, concept of pH and pOH, Buffer solutions, Solubility Product, Redox Reactions.

Electrode Potential, electrochemical cell, concentration cell, reference Electrodes, Overvoltage, Conductometric Titrations, Fuel cells, Lead Acid Storage Cell and numericals based on the above articles.

Unit VI **08 Hours**

STEREOCHEMISTRY

Introduction, chirality, optical activity, Enantiomers, Diastereomers, projection formula of tetrahedral carbon- Newman projection, Wedge projection, Fischer projection, Geometrical isomerism :- cis and trans isomerism, E and Z isomers

Optical isomerism :- Mesoform, the number of optical isomers for chiral molecules, Conformations :- conformations of ethane, conformations of n-butane

Term Work:

Any Ten experiments from the following:

1. Estimation of hardness of water by EDTA method.
2. Estimation of chlorine by Mohr's method.
3. Determination of percentage of Ca in given cement sample
4. Determination of coefficient of viscosity by Ostwald's viscometer
5. Study of Bomb calorimeter for determination of calorific value.

6. Determination of calorific value of gas fuel by using Boy's gas calorimeter.
7. Determination of dissolved oxygen in a water sample.
8. To determine the Molecular Weight of polymer
9. Estimation of Copper from brass sample solution by Iodometrically
10. Estimation of percentage of Iron in Plain Carbon Steel by Volumetric Method
11. To standardize NaOH solution and hence find out the strength of given hydrochloric Acid solution
12. To determine Surface Tension of given liquid by Stalagmometer
13. Study of corrosion of metals in medium of different pH.
14. To set up Daniel cell
15. To determine pH of soil
16. To determine Acidity of soil

Assignment:

7. Effect of hard water on boilers and heat exchangers
8. Hydraulic/ Non-hydraulic cementing materials
9. Analysis of coal a) Proximate b) ultimate analysis of coal
10. Wet corrosion-mechanism, Electroplating, Hot dipping
11. Geometrical isomerism :- cis and trans isomerism, E and Z isomers
12. Fuel cells

Reference Books:

7. Engineering Chemistry by Jain and Jain, Dhanpat Rai Company (P) Ltd, New Delhi
8. Chemistry of Engineering Materials, Agarwal C.V, Rata Publication Varanasi, 6th edition (1979)
9. Chemistry in Engineering and Technology, Volume W, Tata McGraw Hill Publishing Company Ltd, New Delhi (1988)
10. Applied Chemistry, O. P. Vidyankar, J. Publications, Madurai, (1955)
11. Engineering Chemistry, S. N. Chand and Co., Jalandhar, 31st Edition (1990)
12. Engineering Chemistry by Dara S. S. S Chand Publications
13. Fundamentals of Electrochemistry, V. S. Bagotsky (Ed) Wiley NY (2006)

Syllabus for Unit Tests:

Unit Test I :- Unit I,II,III

Unit Test II :- Unit IV,V,VI

02: Fundamentals of Electrical Engineering

Teaching Scheme	Examination Scheme	Credit Scheme
Theory: 04 Hours / Week	End Semester Examination: 60 Marks	03 Credits
Practical: 02 Hours / Week	Continuous Assessment: 40 Marks	
	Term Work: 25 Marks	01 Credit

Course Pre-requisites:

The Students should have

1. Mathematics
2. Physics

Course Objectives:

The course introduces fundamental concepts of DC and AC circuits, electromagnetism, transformer and measuring instruments and electronic components to all fist year engineering students.

Course Outcomes:

1. Understand and apply knowledge of basic concepts of work ,power ,energy for electrical, mechanical and thermal systems
2. Understand and apply knowledge of Kirchoff's laws and network theorems to solve electricalnetworks
3. Describe construction, principle of operation, specifications and applications of capacitors and batteries
4. Describe and apply fundamental concepts of magnetic and electromagnetic circuits for operation of single phase transformer
5. Define basic terms of single phase and three phase ac circuits and supply systems
6. Know and use electrical safety rules

Unit I **06 Hours**

Basic concepts .

Concept of EMF, Potential Difference, current, resistance, Ohms law, resistance temperature coefficient, SI units of Work, power, energy. Conversion of energy from one form to another in electrical, mechanical and thermal systems

Unit II **06 Hours**

Network Theorems

Voltage source and current sources, ideal and practical, Kirchoff's laws and applications to network solutions using mesh analysis, Simplifications of networks using series- parallel, Star/Delta transformation. Superposition theorem, Thevenin's theorem, Max Power Transfer theorem.

Unit III **06 Hours**

Electrostatics

Electrostatic field, electric field intensity, electric field strength, absolute permittivity, relative permittivity, capacitor composite, dielectric capacitors, capacitors in series& parallel, energy stored in capacitors, charging and discharging of capacitors, Batteries-Types, Construction& working.

Unit IV **06 Hours**

Magnetic Circuit & Transformer

Magnetic effect of electric current, cross and dot convention, right hand thumb rule, concept of flux, flux linkages, Flux Density, Magnetic field, magnetic field strength, magnetic field intensity, absolute permeability, relative permeability, B-H curve, hysteresis loop, series-parallel magnetic circuit, composite magnetic circuit, Comparison of electrical and magnetic circuit Farady's law of electromagnetic induction, statically and dynamically induced emf, self inductance, mutual inductance, coefficient of coupling, Single phase transformer construction, principle of operation, EMF equation, voltage ratio, current ratio, kVA rating, losses in transformer, Determination of Efficiency & Regulation by direct load test.

Unit V **06 Hours**

AC Fundamentals & AC Circuits

AC waveform definitions , form factor, peak factor, study of R-L, R-C, RLC series circuit, R-L-C parallel circuit, phasor representation in polar & rectangular form, concept of impedance, admittance, active, reactive, apparent and complex power, power factor, 3-ph AC Circuits.

Unit VI **06 Hours**

Electrical Wiring and Illumination system

Basic layout of distribution system, Types of Wiring System & Wiring Accessories, Necessity of earthing, Types of earthing, Different types of lamps (Incandescent, Fluorescent, Sodium Vapour, Mercury Vapour, Metal Halide, CFL, LED), Study of Electricity bill.

Term Work:

The term work shall consist of record of minimum eight exercises / experiments.

1. Determination of resistance temperature coefficient
2. Verification of Superposition Theorem
3. Verification of Thevenin's Theorem
4. Verification of Kirchoff's Laws
5. Verification of Maximum power transfer Theorem
6. Time response of RC circuit
7. Study of R-L-C series circuits for XL> XC , XL< XC & XL= XC
8. Verification of current relations in three phase balanced star and delta connected loads.
9. Direct loading test on Single phase transformer a)Voltage and current ratios.
   b) Efficiency and regulations .
10. Study of a Residential (L.T.) Bill

Textbooks

1) B.L.Theraja- "A Textbook of Electrical Technology" Volume- I, S.Chand and Company Ltd.,New Delhi
2) V. K. Mehta, - "Basic Electrical Engineering", S. Chand and Company Ltd., New Delhi
3) I. J. Nagrath and Kothari – "Theory and problems of Basic Electrical Engineering", Prentice Hall of India Pvt. Ltd

Reference Books

1. Edward Hughes – "Electrical Technology"- Seventh Edition, Pearson Education Publication
2. H. Cotton – "Elements of Electrical Technology", C.B.S. Publications
3. John Omalley Shawn – "Basic circuits analysis" Mc Graw Hill Publications
4. Vincent Del Toro – "Principles of Electrical Engineering", PHI Publications

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

ENGINEERING MATHEMATICS-II

Teaching Scheme	**Examination Scheme**	**Credit Scheme**
Lectures: 3Hrs/Week	End Semester Examination: 60 marks	Theory : 03
Tutorials: 1Hr/Week	Continuous Assessment:40 marks	Tutorial : 01

Unit I **08 Hours**

DIFFERENTIAL EQUATIONS (DE)

Definition, Order and Degree of DE, Formation of DE.Solutions ofVariable Separable DE, Exact DE, Linear DE and reducible to these types.

Unit II **08 Hours**

APPLICATIONS OF DIFFERENTIAL EQUATIONS

Applications of DE to Orthogonal Trajectories, Newton's Law of Cooling, Kirchoff’s Law of Electrical Circuits, Motion under Gravity, Rectilinear Motion, Simple Harmonic Motion, One–Dimensional Conduction of Heat, Chemical engineering problems.

Unit III **08 Hours**

FOURIER SERIES

Definition, Dirichlet's conditions, Fourier Series and Half Range Fourier Series,Harmonic Analysis.

INTEGRAL CALCULUS

Reduction formulae, Beta and Gamma functions.

Unit IV **08 Hours**

INTEGRAL CALCULUS

Differentiation Under the Integral Sign, Error functions.

CURVE TRACING

Tracing of Curves, Cartesian, Pola and Parametric Curves.Rectification of Curves.

Unit V **08 Hours**

SOLID GEOMETRY

Cartesian, Spherical Polar and Cylindrical Coordinate Systems. Sphere, Cone and Cylinder.

Unit VI **08 Hours**

MULTIPLE INTEGRALS AND THEIR APPLICATIONS

Double and Triple integrations, Applications to Area,Volume, Mean and Root Mean Square Values.

Assignment:

1. Differential Equations.
2. Application of DE.
3. Fourier Series and Integral Calculus.
4. DUIS and Curve Tracing.
5. Solid Geometry.
6. Double and Triple integrations, area and volume.

Textbooks:

1. Advanced Engineering Mathematics by Erwin Kreyszig, Wiley Eastern Ltd, 8thedition(1999).
2. Higher Engineering Mathematics by B.V. Ramana, Tata McGraw-Hill (2008)
3. Applied Mathematics (Volumes I and II) by P. N. Wartikar& J. N. Wartikar, Pune VidyarthiGrihaPrakashan, Pune, 7th edition (1988).
4. Higher Engineering Mathematics by B. S. Grewal, Khanna Publication, Delhi, 42th edition (2012).
5. Advanced Engineering Mathematics, 7e, by Peter V. O'Neil, Thomson Learning,6th edition (2007).
6. Advanced Engineering Mathematics, 2e, by M. D. Greenberg, Pearson Education, 2nd edition(2002).

Syllabus for Unit Test:

Unit Test I :-Unit I,II,III

Unit Test II :-Unit IV,V,VI

FUNDAMENTALS OF MECHANICAL ENGINEERING

Teaching Scheme	Examination Scheme	Credit Scheme
Theory: -03Hours / Week	End Semester Examination: -60Marks	04
Practical: 02 Hours / Week	Continuous Assessment: -40Marks	
	Term Work: 25 Marks	

Unit I **08 Hours**

Thermodynamics-

Heat, work and Internal Energy, Thermodynamic State, Process, Cycle, Thermodynamic System, First Law of Thermodynamics, Application of First Law to steady Flow and Non Flow processes, Limitations of First Law, PMM of first kind (Numerical Treatment), Second Law of Thermodynamics – Statements, Carnot Engine and Carnot Refrigerator, PMM of Second Kind (Elementary treatment only)

Unit II **06 Hours**

Introduction to I.C. Engines and turbines-

Two stroke, Four Stroke Cycles, Construction and Working of C.I. and S.I. Engines, Hydraulic turbines, steam turbines, gas turbines.(Theoretical study using schematic diagrams)

Introduction to refrigeration, compressors & pumps-

Vapor compression and vapor absorption system, house hold refrigerator, window air conditioner. Reciprocating and rotary compressor, Reciprocating and centrifugal pump. (Theoretical study using schematic diagrams)

Unit III **06 Hours**

Energy Sources -

Renewable and nonrenewable, solar flat plate collector, Wind, Geothermal, Wave, Tidal, Hydro power, Bio-gas, Bio-Diesel, Nuclear power.

Heat transfer-

Statement and explanation of Fourier's law of heat conduction, Newton's law of cooling, Stefan Boltzmann's law. Conducting and insulating materials and their properties, types of heat exchangers and their applications.

Unit IV **06 Hours**

Properties of fluids-

Introduction, Units of measurements, mass density, specific weight, specific volume and relative density, viscosity, pressure, compressibility and elasticity, gas laws, vapor pressure, surface tension and capillarity, regimes in fluid mechanics, fluid properties and analysis of fluid flow.

Properties of Materials and their Applications-

Metals – Ferrous and Non-Ferrous, Nonmetallic materials, smart materials, Material selection criteria.

Unit V **06 Hours**

Mechanical devices -

Types of Belts and belt drives, Chain drive, Types of gears, Types of Couplings, friction clutch (cone and plate), brakes, Power transmission shafts, axles, keys, bush and ball bearings.

Mechanisms-

Slider crank mechanism, Four bar chain mechanism, List of various inversions of Four bar chain mechanism, Geneva mechanism, Ratchet and Paul mechanism

Unit VI **06 Hours**

Machine Tools-

Lathe Machine – Centre Lathe, Drilling Machine – Study of Pillar drilling machine, Introduction to NC and CNC machines, Grinding machine, Power saw, Milling Machine.

Introduction to manufacturing processes and Their Applications-

Casting, Sheet metal forming, Sheet metal cutting, Forging, Fabrication, Metal joining processes.

List of experiments-

The Term Work shall consist of any Eight experiments of following list

1 Measurement of viscosity using Redwood viscometer.

2 Assembly and working of 4-bar, 6-bar, 8-bar planer mechanisms

3 Finding relation between input angle and output angle for various link lengths.

4 Study of domestic refrigerator & window air conditioner

5 Demonstration of operations of centre lathe

6 Demonstration of operations on drilling machines

7 Demonstration of Two stroke and four stroke engine

8 Study of power transmitting elements: Coupling, Gears and bearings

9 Demonstration of pumps and compressor

10 Study and demonstration of different types of clutches.

Reference Books:

1 "Thermodynamics An Engineering Approach" Yunus A. Cengel and Michael A. Boles, McGraw-Hill, Inc,2005,6th edition.

2. "Applied Thermodynamics for Engineering Technologists" T. D. Eastop and A. McConkey, 5th Edition, Prentice Hall.

3. "I.C. Engines Fundamentals" J. B. Heywood, McGraw Hill, 3rd Edition, MacMillian

4. "Internal Combustion Engine ": V. Ganeshan, Tata McGraw-Hill, 3rd edition. 5 "Strength of Materials" H. Ryder, Macmillians, London, 1969, 3rd edition.

6. "Mechanics of Materials" Johston and Beer TMH, 5th edition

7 "Mechanisms and Machine Theory" Ambekar A.G., Prentice-Hall of India, 2007.

8. "Theory of Machines" S.S. Rattan, Tata McGraw- Hill, 2nd edition.

9 "A Textbook of production engineering" P.C. Sharma, S. Chand Publication, New Delhi, 2nd edition.

10 "Fluid Mechanics & Fluid Power" D.S. Kumar, Katson Publishing Engineering House, Ludhiana. 8th edition

10: Engineering Mechanics

Teaching Scheme	Examination Scheme	Credit Scheme
Theory: 04 Hours / Week	End Semester Examination: 60 Marks	03 Credits
Practical: 02 Hours / Week	Continuous Assessment: 40 Marks	
	Term Work: 25 Marks	01 Credit

Course Pre-requisites:

The Students should have knowledge of

1. Scalar and Vector
2. Newton's law of motion
3. Law of friction
4. Concept of physical quantities, their units and conversion of units
5. Concept of differentiation and integration

Course Objectives:

To develop and apply the concept of resultant and equilibrium for various static and dynamic engineering problems.

Course Outcomes: The student should be able to

1. calculate resultant and apply conditions of equilibrium.
2. analyze the truss and calculate friction force.
3. calculate centroid and moment of inertia.
4. solve problem on rectilinear motion.
5. solve problems on curvilinear motion.
6. useD'Alembert's principle, Work Energy principle and Impulse Momentum principle for particle.

Unit I **06 Hours**

Resultant and Equilibrium

Types and Resolution of forces, Moment and Couple, Free Body Diagram, Types of Supports, Classification and Resultant of a force system in a Plane - Analytical and Graphical approach..

Equilibrant, Conditions of Equilibrium, Equilibrium of a force system in a Plane,Force and Couple system about a point.

Unit II **06 Hours**

Truss and Friction

Coefficient of Static Friction, Impending motion of Blocks, Ladders and Belts.

Analysis of Perfect Trusses - Method of Joint, Method of Section and Graphical Method.

Unit III **06 Hours**

Centroid and Moment of Inertia

Centroid of line and plane areas, Moment of Inertia of plane areas, parallel and perpendicular axis theorem, radius of gyration, least moment of inertia.

Unit IV **06 Hours**

Kinematics of Rectilinear motionof a Particle

Equations of motion, Constant and variable acceleration, Motion Curves, Relative motion, Dependent motion.

Unit V **06 Hours**

Kinematics of Curvilinear motionof a Particle

Motion of a Projectile, Cartesian components, Normal and Tangential components of a curvilinear motion.

Unit VI **06 Hours**

Kinetics of a Particle

D'Alemberts Principle, Work-Energy Principle and Impulse-Momentum Principle, Coefficient of Restitution, Direct Central Impact.

Term Work:

A) The term-work shall consist of minimum Five experiments from list below.

1. Determination of reactions of Simple and Compound beam.
2. Study of equilibrium of concurrent force system in a plane.
3. Determination of coefficient of friction for Flat Belt.
4. Determination of coefficient of friction for Rope.
5. Study of Curvilinear motion.
6. Determination of Coefficient of Restitution.

B) The term-work shall also consist of minimum Five graphical solutions of the problems on different topics.

Text Books:

1) "Engineering Mechanics (Statics and Dynamics)",Hibbeler R.C., McMillan Publication.
2) "Vector Mechanics for Engineers-Vol.-I and Vol.-II (Statics and Dynamics)",Beer F.P. and Johnston E.R., Tata McGraw Hill Publication.
3) "Engineering Mechanics",Bhavikatti S.S. and Rajashekarappa K.G., New Age International (P) Ltd.

Reference Books:

1. "Engineering Mechanics (Statics and Dynamics)", Shames I.H., Prentice Hall of India (P) Ltd.
2. "Engineering Mechanics (Statics and Dynamics)", Singer F.L., Harper and Row Publication.
3. "Engineering Mechanics (Statics and Dynamics)",Meriam J.L. and Kraige L.G., John Wiley and Sons Publication.
4. "Engineering Mechanics (Statics and Dynamics)",Timoshenko S.P. and Young D.H., McGraw Hill Publication.
5. "Engineering Mechanics (Statics and Dynamics)",Tayal A.K., Umesh Publication.
6. "Engineering Mechanics-I and II (Statics and Dynamics)", Mokashi V.S., Tata McGraw Hill Publication.

Syllabus for Unit Tests:

Unit Test -1 UNIT – I to III

Unit Test -2 UNIT – IV to VI

ENGINEERING PHYSICS

Teaching Scheme	Examination Scheme	Credit Scheme
Lectures: 4Hrs/Week	End Semester Examination: 60 marks	Theory: 04
Practical: 2Hr/Week	Continuous Assessment:40 marks	Practical: 01
	Term Work: 25marks	

Unit I **08 Hours**

MODERN PHYSICS

Motion of a charged particle in electric and magnetic fields, Electrostatic and Magnetostatic focussing, Wavelength and resolution, Specimen limitation, Depth of field and focus, Electron microscope, Positive rays, Separation of isotopes by Bainbridge mass spectrograph.

NUCLEAR PHYSICS

Nuclear fission, Liquid drop model of nucleus, Nuclear fission in natural uranium, Fission energy, Critical mass and size, Reproduction factor, Chain reaction and four factor formula, Nuclear fuel and power reactor, Nuclear fusion and thermonuclear reactions, Merits and demerits of nuclear energy, Particle accelerators, Cyclotron, Betatron,

Unit II **08 Hours**

SOLID STATE PHYSICS

Band theory of solids, Free electron theory, Fermi-Dirac probability function and position of Fermi level in intrinsic semi-conductors (with derivation) and in extrinsic semi-conductors, Band structure of p-n junction diode under forward and reverse biasing, Conductivity in conductor and semi-conductor, Hall effect and Hall coefficient, Photovoltaic effect, Solar cell and its characteristics.

SUPER CONDUCTIVITY

Introduction, Properties of a super conductor, Meissner's effect, Critical field, Types of superconductors, BCS theory, High temperature superconductors, Application of superconductors.

Unit III **08 Hours**

THERMODYNAMICS

Zeroth law of thermodynamics, first law of thermodynamics, determination of j by Joule's method, Applications of first law, heat engines, Carnot's cycle and Carnot's engine, second law of thermodynamics, entropy, change in entropy in reversible and irreversible processes, third law of thermodynamics.

NANOSCIENCE

Introductions of nanoparticals, properties of nanoparticals (Optical, electrical, Magnetic, structural, mechanical), synthesis of nanoparticals(Physicaland chemical), synthesis of clloids, growth of nanoparticals, synthesis of nanoparticals by colloidal rout, applications.

Unit IV **08 Hours**

OPTICS - I

INTERFERENCE

Interference of waves, Visibility of fringes, interference due to thin film of uniform and non-uniform thickness, Newton's rings, Engineering applications of interference (optical flatness, interference filter, non-reflecting coatings, multi-layer ARC.

DIFFRACTION

Classes of diffraction, Diffraction at a single slit (Geometrical method), Conditions for maximum and minimum, Diffraction at a circular aperture (Result only), Plane diffraction grating, Conditions for principal maxima and minima, Rayleigh's criterion for resolution, Resolving power of grating and telescope.

Unit V **08 Hours**

POLARISATION

Introduction, Double refraction and Huygen's theory, Positive and negative crystals, Nicol prism, Dichroism, Polaroids, Elliptical and circular polarisation, Quarter and half wave plates, Production of polarised light, Analysis of polarised light, half shade polarimeter, LCD.

LASERS

Spontaneous and stimulated emission, Population inversion, Ruby laser, Helium-Neon laser, Semiconductor laser, Properties of lasers, Applications of lasers (Engineering/ industry, medicine, communication, Computers), Holography.

Unit VI **08 Hours**

ARCHITECTURAL ACOUSTICS

Elementary acoustics, Limits of audibility, Reverberation and reverberation time, Sabine's formula, Intensity level, Sound intensity level, Sound absorption, Sound absorption coefficient, different types of noise and their remedies, Sound absorption materials, basic requirement for acoustically good hall, factors affecting the architectural acoustics and their remedies.

QUANTUM MECHANICS

Electron diffraction, Davisson and Germer's experiment, Wave nature of matter, De-Broglie waves, Wavelength of matter waves, Physical significance of wave function, Schrodinger's time dependant and time independent wave equation, Application of Schrodinger's time independent wave equation to the problems of Particle in a rigid box and non rigid box.

TERM WORK

Experiments

Any ten experiments from the following:

1. Determination of band gap of semi-conductor.
2. Solar cell characteristics.
3. e/m by Thomson's method.
4. Uses of CRO for measurement of phase difference and Lissajos figures.
5. Hall effect and Hall coefficient.
6. Conductivity by four probe method.
7. Diode characteristics (Zener diode, Photo diode, LED, Ge/Si diode).
8. Plank's constant by photodiode.
9. Wavelength by diffraction grating.
10. Newton's rings.

11. Ultrasonic interferometer.
12. Sound intensity level measurement.
13. Wavelength of laser by diffraction.
14. Determination of refractive index for O-ray and E-ray.
15. Brewester’s law.

11. Ultrasonic interferometer.
12. Sound intensity level measurement.
13. Wavelength of laser by diffraction.
14. Determination of refractive index for O-ray and E-ray.
15. Brewester’s law.

Assignments

1. Recent advances in Nanotechnology
2. Nuclear radiation detectors.
3. Atomic force microscope (AFM).
4. Advanced opto-electronic devices.
5. Laser in Industry.
6. Different spectroscopic methods – a comparison (Raman, IR, UVR, etc.).

Reference Books:

1. Physics for Engineers – Srinivasan M.R.
2. A text Book of Engineering Physics- M.N. Avadhanulu, P.G. Kshirsagar
3. Engineering Physics- K. Rajagopal
4. Electronics Principles – A.P.Molvino
5. Fundamentals of Optics – Jenkins and White
6. A Textbook of Sound – Wood
7. Engineering Physics – Sen, Gaur and Gupta

Syllabus for Unit Tests:

Unit Test I : Unit I, II, III

Unit Test II: Unit IV, V, VI

Object-Oriented Programming in 'C++'

Teaching Scheme	Examination Scheme	Credit Scheme
Lectures : 2 Hrs/week	ESE (End Semester Examination) : 60 Marks	
	Continuous Assessment: 40 Marks	
Practicals : 2 Hrs/week	Term Work : 25 Marks	Credits : 3 Credits

Course Pre-requisites:

Programming Principles and Paradigms, "C" programming language.

Course Objectives:

- To familiarize with the universal concepts of computer programming.
- To present the syntax and semantics of the "C++" language as well as basic data types offered by the language
- To discuss the principles of the object-oriented model and its implementation in the "C++" language
- To demonstrate the means useful in resolving typical implementation problems with the help of standard "C++" language libraries

Course Outcomes:

At the end of this course students will able to :

- Understand basic concepts of Object Oriented Programming and applications of OOP.
- Usebasic, user-defined and derived data types, Operator precedence.
- Apply Decision Structure, Loops and Functions
- Write, Debug and Compile Programs of C++
- Implement OOP concepts like – Inheritance using C++ programming.

Unit I **03 Hours**

Principles of Object Oriented Programming:

Object Oriented Programming Paradigm, Basic concepts of Object Oriented Programming, Benefits of OOP, Object Oriented Languages, Applications of OOP.

Unit II **05 Hours**

Beginning with C++:

Overview of C++, Sample C++ Program, C++ statements, Structure of C++ program,Creating source file , compiling and Linking, Tokens, Keywords , Identifiers and Constants, Basic data types, User-defined data types, Derived data types, Declaration of variables, Dynamic initialization of variables, Scope Resolution Operator, Operator Overloading, Operator precedence, Control Structures.

Unit III **05 Hours**

Functions in C++:

The Main Function, Function Prototyping, Call by Reference, Inline functions, Default arguments, Function Overloading , Friend and Virtual Functions .

Classes and Objects:

Class specification , Class Objects , Scope resolution operator, Accessspecifiers-Public,Private,Protected,Defining member Functions, Nesting of Member Functions,Private Member Functions, Static Data Members , Static Member Functions, Data hiding.

Unit IV **06 Hours**

Inheritance: Extending Classes:

Defining Derived Classes, Single Inheritance, Making a Private member inheritable, Multilevel Inheritance, Multiple Inheritance, Hierarchical Inheritance,Hybrid Inheritance, Virtual Base Class, Abstract class. Inheritance and protected members, Protected base class inheritance, Inheriting multiple base classes, Constructors, Destructors , Passing parameters to base class constructors, virtual base classes.STL: An overview, containers, vectors, lists, maps.

Unit V **06 Hours**

Constructors and Destructors:

Constructors,Parameterized constructors, Default Constructors, Copy constructor, Dynamic Initialization of Objects, Destructors.

Polymorphism:

Base class, Virtual Functions, Pure Virtual Functions,Calling a virtual function through a base classreference, Early and Late Binding.

Unit VI **03 Hours**

Managing Console I/O operations:

C++ Stream Classes, Unformatted I/O Operations, Working with Files, Opening and Closing afile, Formatted I/O.

Textbooks:

- Herbert Schildt, "The Complete Reference C++", 4thEdition, Mc Graw Hill, 2003.
- Stanley.B.Lippmann, Josee Lajoie, Barbara.E.Moo, "C++ Primer", 5th Edition, Pearson Education, 2013.
- Scott Meyers:"Effective C++",Third Edition, Addison-Wesley, 2005.
- E. Balaguruswamy, "Object Oriented Programming using C++", 4th Edition,Mc Graw Hill, 2010.

Syllabus for Unit Test

Unit Test 1	Unit I & II
Unit Test 2	Unit III & IV
Unit Test 3	Unit V & VI

Workshop Technology

Teaching Scheme	Examination Scheme	Credit Scheme
Theory: -	End Semester Examination: -	
Practical: 02 Hours / Week	Continuous Assessment: -	
	Term Work: 50 Marks	

Course Pre-requisites:

Basic knowledge of hand tools used in day to day life.

Course Objectives:

Make the students familiar with basic manufacturing processes

Course Outcomes: students should be able to understand

1. basic Manufacturing Processes used in the industry,
2. importance of safety

Term work shall consist of any three jobs, demonstrations on rest of the trades and journal consisting of six assignments one on each of the following topics.

Carpentry- Introduction to wood working, kinds of woods, hand tools & machines, Types of joints, wood turning. Pattern making, types of patterns, contraction, draft & machining allowances

Term work includes one job involving joint and woodturning.

Fitting- Types of Fits, concepts of interchangeability, datum selection, location layout, marking, cutting, shearing, chipping, sizing of metals, drilling and tapping.

Term work to include one job involving fitting to size, male-female fitting with drilling and tapping.

Sheet Metal Practice Introduction to primary technology processes involving bending, punching and drawing various sheet metal joints, development of joints.

Joining- Includes making temporary and permanent joints between similar and dissimilar material by processes of chemical bonding, mechanical fasteners and fusion technologies.

Term work includes one job involving various joining processes like riveting, joining of plastics, welding, brazing, etc.

Forging -Hot working, cold working processes, forging materials, hand tools & appliances, Hand forging, Power Forging.

Moulding -Principles of moulding, methods, core & core boxes, preparation of foundry sand, casting, Plastic moulding.

Plumbing (Demonstration Common for Electrical & Non electrical Group)

Types of pipe joints, threading dies, Pipe fittings.

BHARATI VIDYAPEETH
(DEEMED TO BE UNIVERSITY) Pune.

Faculty of Engineering & Technology
B.Tech. (Computer Science and Engineering)
Program Curriculum
(2014 Course)

COURSE STRUCTURE AND SYLLABUS
(Choice Based Credit System - 2014 Course)
B.Tech. (Computer Engineering)
Program Curriculum Sem V & VI

Programme: B. Tech.Computer Engineering

VISION OF UNIVERSITY:

Social Transformation through Dynamic Education

MISSION OF UNIVERSITY:

- To make available quality education in different areas of knowledge to the students as per their choice and inclination.
- To offer education to the students in a conducive ambience created by enriched infrastructure and academic facilities in its campuses.
- To bring education within the reach of rural, tribal and girl students by providing them substantive fee concessions and subsidized hostel and mess facilities.
- To make available quality education to the students of rural, tribal and other deprived sections of the population.

VISION OF THE INSTITUTE:

To be World Class Institute for Social Transformation Through Dynamic Education.

MISSION OF THE INSTITUTE:

- To provide quality technical education with advanced equipment, qualified faculty members, infrastructure to meet needs of profession and society.
- To provide an environment conducive to innovation, creativity, research, and entrepreneurial leadership.
- To practice and promote professional ethics, transparency and accountability for social community, economic and environmental conditions.

VISION OF THE DEPARTMENT

To be focused on innovative and quality education in computer science and engineering that prepares professionals for development of society.

MISSION OF THE DEPARTMENT

- To provide academic environment for the development of skilled professionals
- To cultivate research culture that contributes to the sustainable development of the society.
- To enhance academic and industry collaborations for global exposure.

Programme: B. Tech.Computer Engineering

PROGRAM EDUCATIONAL OBJECTIVES

The students of B.TECH. (Computer Science and Engineering), after graduating will able to,

1. Demonstrate technical and professional competencies by applying Engineering Fundamentals, knowledge of computing and technologies.
2. Exhibit effective personality, good communication and team building skills
3. Adopt to the latest trends in the field of computer science and engineering.

PROGRAM SPECIFIC OUTCOMES

1. To design, develop and implement computer programs on hardware towards solving problems.
2. To employ expertise and ethical practice through continuing intellectual growth and adapting to the working environment.

PROGRAM OUTCOMES

a. Apply the knowledge of mathematics, science, engineering, and computing to provide a solution of complex engineering problems.

b. Identify, analyse complex engineering problems to derive conclusions using computer science and engineering knowledge.

c. Outline resolutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration, societal, and environmental considerations.

d. Use existing research knowledge and research techniques including design of experiments, data analysis, and synthesis to provide valid inferences.

e. Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools.

f. Apply inferences obtained by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the subsequent responsibilities relevant to the professional engineering practice.

g. Recognize the impact of the professional engineering solutions in societal and environmental contexts to demonstrate the knowledge for sustainable development.

h. Apply ethical principles and execute professional ethics and responsibilities and norms of the engineering practice.

i. Work effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary surroundings.

j. Talk effectively on complex engineering activities with the engineering community such as being able to comprehend and write effective reports and design documentation, make effective presentations.

k. Prove knowledge and understanding of the engineering and management principles and apply these to one's work, as a member and leader in a team.

l. Recognise the need for and have the preparation and ability to engage in independent and life-long learning in context of technological change.

CORELATION BETWEEN GRADUATE ATTRIBUTES AND PROGRAMME OUTCOMES

Graduate Attributes/ Programme Outcomes	a	b	c	d	e	f	g	h	i	j	k	l
Engineering Knowledge	✓											
Problem Analysis		✓										
Design/Development of solutions			✓									
Conduct Investigations of Complex Problems				✓								
Modern Tool Usage					✓							
The Engineer and Society						✓						
Environment and Sustainability							✓					
Ethics								✓				
Individual and teamwork									✓			
Communication										✓		
Project management and finance											✓	
Life-long learning												✓

Bharati Vidyapeeth University, Pune
Faculty of Engineering & Technology
Programme: B. Tech (Computer Engineering) Sem – V (2014 Course)

Sr. No	Name of Course	Teaching Scheme			Examination Scheme							Credits		
		L	P	T	ESE	Continuous Assessment			Practical		Total	Theory	TW	Total
						Unit Test	Attendance	Assignment	TWPR	TW OR				
1	Formal Language and Automata Theory	3	--	1	60	20	10	10	--	--	100	4	-	4
2	Software Testing	3	2	--	60	20	10	10	--	50	150	3	1	4
3	Computer Network	3	2	--	60	20	10	10	50	--	150	3	1	4
4	Database Management System	3	2	--	60	20	10	10	50	--	150	3	1	4
5	Elective - I	3	--	--	60	20	10	10	--	--	100	3	--	3
6	Professional Skill Development-V	4	--	--	100	--	--	--	--	--	100	4	--	4
7	Programming Lab-III	--	4	--	--	--	--	--	50	--	50	--	2	2
	TOTAL	**19**	**10**	**1**	**400**	**100**	**50**	**50**	**150**	**50**	**800**	**20**	**05**	**25**

Elective - I: a) Multimedia and Mobile Applications b) Scripting Languages c) Software Project Management d) Computational Genomics

Optional Subject

Sr. No	Name of Course	Teaching Scheme			Examination Scheme							Credits		
		L	P	T	ESE	Continuous Assessment			Practical		Total	Theory	TW	Total
						Unit Test	Attendance	Assignment	TW PR	TW OR				
	Engineering Mathematics IV	4	--	--	60	20	10	10	--	--	100	4	--	4

Bharati Vidyapeeth University, Pune
Faculty of Engineering & Technology
Programme: B. Tech (Computer Engineering) Sem –VI (2014 Course)

Sr. No.	Name of Course	Teaching Scheme			Examination Scheme							Credits		
		L	P	T	ESE	Continuous Assessment			Practical		Total	Theory	TW	Total
						Unit Test	Attendance	Assignment	TW PR	TW OR				
8	Operating System	3	2	--	60	20	10	10	--	50	150	3	1	4
9	Design and Analysis of Algorithm	3	2	--	60	20	10	10	--	50	150	3	1	4
10	Digital Signal Processing	3	2	--	60	20	10	10	--	50	150	3	1	4
11	Computer Organization and Architecture	3	--	1	60	20	10	10	--	--	100	4	--	4
12	Elective - II	3	--	--	60	20	10	10	--	--	100	3	--	3
13	Professional Skill Development-VI	4	--	--	100	--	--	-	--	--	100	4	--	4
14	Programming Lab-IV	--	4	--	--	--	--	--	50	--	50	--	2	2
	TOTAL	**19**	**10**	**1**	**400**	**100**	**50**	**50**	**150**	**50**	**800**	**20**	**05**	**25**

Elective - II: a) VLSI b) Natural Language Processing c) Human Computer Interaction d) Data Storage Systems

Total Credits Sem – III : 25
Total Credits Sem – IV : 25
Grant total : 50

FORMAL LANGAUGE AND AUTOMATA THEORY

Teaching Scheme	Examination Scheme	Credit Scheme
Theory: 3 Hours / Week	End Semester Examination: 60 Marks	Theory:04 Credits
Tutorial: 1 Hours / Week	Continuous Assessment: 40 Marks	

Course Pre-requisites:

1. Discrete Mathematics
2. Introductions to algorithms

Course Objectives:

1. Learn types of grammars.
2. Understand various Computing models like Finite State Machine, Pushdown Automata, and
3. Turing Machine.
4. To Learn various applications of Formal Language Processing

Course Outcomes: After completion of course, students will able to:

1. Illustrate the importance of Automata Theory in designing computer languages.
2. Transform informal problems into formal ones.
3. Infer Grammars, languages modeling and compilers basics.
4. Investigate and prove the equivalence of languages described by pushdown automata
5. Design Language Acceptability by Turing Machine
6. Outline the applicability of the formal language and automata theory concepts.

Unit I **06 Hours**

Finite State Machine (FSM): Introduction, difference between natural and formal languages, Basic machine - design of basic machines. Transition diagram, Transition graph, Acceptance of String, Acceptance of Language, examples, Finite Automata (FA), Deterministic Finite Automaton (DFA) and Non-Deterministic Finite Automaton (NDFA), Conversions of NFA with Є and without Є, Minimization of DFA, Equivalence of NFA and DFA, Limitations of FA.

Unit II **06 Hours**

Regular expressions (RE) - Introduction, FA and RE, RE to FA, FA to RE, algebraic laws, applications of REs, FA for regular grammar, Uses of Regular expression, Pumping Lemma.

Moore and Mealy Machines: Introduction, Difference between Moore & Mealy, models, inter conversions, Equivalence of Mealy machine and Moore machine, Uses of both the machines.

Unit III **06 Hours**

Grammar- Introduction, representation of grammar, Chomsky hierarchy, Context Free Grammar- Derivation, sentential form, inference, derivation, parse tree, ambiguity in grammar and language- ambiguous Grammar, removing ambiguity from grammar, Normal Forms- Chomsky normal form, Greibach normal form, Closure properties of CFL, Decision property of CFL, Reduced form grammar removal of unit productions, epsilon production, useless symbols. left linear and right linear grammars and inter conversions.

Unit IV **06 Hours**

Push Down Automata (PDA): Introduction, Pushdown Automata (PDA), Transition Diagrams, Functions and Tables, Deterministic Push- down Automata (DPDA) - definition, Nondeterministic Pushdown Automata (NPDA), Equivalence of context free grammars and PDA, properties of context free languages. Introduction to Post Machines (PMs)

Unit V **06 Hours**

Turing Machine (TM): Introduction, Transitions Diagrams, Functions and Tables, Design of TM as generator, decider and acceptor, comparison of Turing machine (TM) with FSM, PDM, and PM. combination TM, iterative TM, recursive TM, universal TM, Language Acceptability by TM, Recursive sets, partial recursive functions, recursively enumerable sets, Church's Turing hypothesis, multi stack Turing machine, TM limitations.

Unit VI **06 Hours**

Applications –Application of RE, FA, PDA, CFG, TM. Syntax analysis language definition. Primitive recursive functions – Recursive and recursively enumerable languages – Universal Turing machine. Lexical analyzer, Text editor, and searching using RE. Introduction to Natural Language Processing.

Assignment:

Note:

For internal Assessment of 10 Marks, students have to submit two assignments based on problems of different types of any programming assignment or theory assignment or any case study or quiz or Multiple Choice Questions etc.

The assignments are to be submitted as a hard copy.

Textbooks:

1. E.V. Krishnamurthy, "Theory of Computer Science", EWP Publication
2. Vivek Kulkarni "Theory Computation" Oxford higher education.

Reference Books:

1. Hopcroft Ullman, "Introduction to Automata Theory, Languages & Computations, Narosa
2. Daniel A. Cohen, "Introduction to Computer Theory", Wiley Publication
3. John C. Martin, "Introduction to Language & Theory of Computation", McGraw Hill
4. Mishra K L P and Chandrasekaran N, "Theory of Computer Science - Automata, Languages and Computation", Third Edition, Prentice Hall of India, 2004

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

Discrete Mathematical Structures

Teaching Scheme	Examination Scheme	Credit Scheme
Theory: 03 Hours / Week	End Semester Exam: 60 Marks	Theory: 03 Credits
Practical: 02 Hours / Week	Continuous Assessment: 40 Marks	TW/OR: 01 Credits
	Term Work/Oral: 50 Marks	

Course Pre-requisites:

The Students should have

1. Basic knowledge of Software Engineering.

Course Objectives:

1. Create awareness among the students about the basic fundamentals of Software Testing Process.
2. To make students aware of various techniques and tools used for testing different kinds of software.

Course Outcomes: After completion of course, students will able to:

1. Define Software Testing level, techniques, test deliverables/artifacts, V-Model.
2. Comprehend various Testing Levels and Techniques.
3. Compare different testing approaches for various kinds of applications.
4. Analyze business and software Risks involved in managing Software Testing Projects.
5. Outline substantial knowledge of Agile Testing Process.
6. Recite Agile Testing Techniques and Tools.

Unit I **06 Hours**

Introduction:

Testing as an Engineering Activity – Basic Definitions – Bug, Defect, Verification validation, Debugging ,Software Testing Principles – The Tester's Role in software Development Organization , Iterative development – Risk Driven and Client Driven, Test driven Development, Evolutionary and adaptive development , Evolutionary requirements analysis – Early "Top Ten" high-level requirements and skillful analysis, Testing Fundamentals- Quality Assurance, Quality Control, V- Model of software testing, Testing techniques and Levels of Testing, Static Vs Dynamic testing, Test deliverables.

Unit II **06 Hours**

Testing Techniques and Test Case Fundamentals.

Testing Levels- Unit Testing, Integration Testing, Functional and System Testing, Types of Testing- Stress Testing, Performance Testing, Usability Testing, Non- Functional Testing, Acceptance Testing, Regression Testing, Beta Testing, Testing techniques- Black Box Vs. White Box Testing, Black Box Techniques, White Box Techniques, and Verification Techniques: Inspection, Walk-Through, Peer Reviews.

Test Case Fundamentals- Definition, Test Case Template, Test Case Parameters, Kinds of Test cases, Characteristics of Good Test Case, Writing Functional and Boundary Test Cases, Manual Test Cases Vs Automated Test Scripts.

Unit III **06 Hours**

Testing approaches for different Types of Applications

Testing COTS (Commercial-Off-the-shelf Software, Web-based applications/e-commerce applications, Object-Oriented Systems, Wireless applications, Testing for security, Website Testing, Foreign Language Testing.

Test Automation Tools: Web browser Automation through Selenium, Test Management Tool-Test Director, Defect Tracking Tool – Bugzilla, GUI Testing Tool- WinRunner, and Configuration Management Tools.

Unit IV **06 Hours**

Managing Software Testing Projects

Test Planning- Test strategy, Creating a test Plan, Test Plan document, Test Pass/Fail criteria, Risk analysis, Test Scoping and Effort Estimation, Test Scheduling and budgeting. Introduction to Testing Maturity Model (TMM).

Test Metrics: Types, Identifying Test Metrics, Methods for gathering Test metrics, Analyzing and Applying Test Metrics, GQM Paradigm. Defect Management: Defect Tracking, Defect Reporting, Defect Metrics.

Unit V **06 Hours**

Agile Methodology

Introduction to Agile Model, Agile Vs Waterfall Method, Methodologies of Agile Testing - Scrum- Practices, Process flow of Scrum, eXtreme Programming-Phases, Crystal Methodologies, DSDM (Dynamic Software Development Method), Agile Testing Lifecycle. Test-Driven Development, Acceptance Test-

Driven Development, and Behavior-Driven Development, Role of the Agile Tester, Assessing Quality Risks in Agile Projects, Techniques in Agile Projects- Acceptance Criteria, Adequate Coverage, Applying Acceptance Test-Driven Development, Exploratory Testing and Agile Testing, Tools in Agile Projects.

Unit VI **06 Hours**

Agile Testing Techniques and Quality Tools:

Agile testing – Nine principles and six concrete practices for testing on agile teams. Six Sigma - Kaoru Ishikawa's Basic Seven QC (Quality Control) Tools- Cause and Effect diagram, Check Sheet, Control charts, Histogram, Pareto chart, Scatter diagram, Flowchart, PDCA (Plan-Do-Check-Act) cycle, Software Testing- ISO Standards, IEEE Standards- IEEE 829, IEEE 1008, IEEE 1012 etc.

Term Work:

1. Introduction to Software Testing Lifecycle Model and the SPRAE Software TestingFramework.
2. Design functional and boundary test cases manually, execute tests, and evaluate test results.
3. Write manual test cases for a C program which demonstrates the working of the following

A. Constructs: I) do...while II) switch

B. A program written in C language for Matrix Multiplication fails." Introspect the causes for its failure and write down the possible reasons for its failure".

4. Write the test cases for any known application (e.g. Banking application)
5. Create a test plan document for any application (e.g. Library Management System)
6. Consider any system (e.g. ATM system) and study its system specifications and report the various bugs.
7. Running test scripts using automated testing tool (e.g. Win runner)
8. Testing of a web application using the web testing tool (e.g. Selenium)
9. Tracking and reporting bugs using bug tracking tool (e.g. Bugzilla, bugbit)
10. Create a database of manual and automated tests using any open source test management tool.
11. Implement software testing on an Agile Project.

Assignments:

1. The Term Work prescribed in the syllabus is continuous assessment by the concerned subject faculty.
2. In case of assignments for internal 10 marks students will be assigned two assignments containing problems of different types or any programming assignment and guided for the solutions of the problem.
3. The assignments are to be submitted as hard copy.

Textbooks:

1. William.E. Perry, "Effective Methods for software Testing", Wiley 3rd Edition.
2. Ron Patton, "Software Testing", Techmedia.
3. Elfriede Dustin," Effective Software Testing", Addison-Wesley, 1st Edition, 2003.
4. Lisa Crispin, Janet Gregory," Agile Testing", Addison-Wesley, 11th Edition, 2015.

Reference Books:

1. Marnie Hutcheson," Software Testing Fundamentals: Methods and Metrics", Wiley.
2. Paul C. Jorgensen, "Software Testing: A Craftsman's Approach", Auerbach Publications, 2008.
3. Craig Larman, "Agile and Iterative Development – A Manager's Guide", Pearson Education – 2004.

Syllabus for Unit Tests:

Unit Test -1 UNIT – I, UNIT – II, UNIT – III

Unit Test -2 UNIT – IV, UNIT – V, UNIT – VI

COMPUTER NETWORK

Teaching Scheme	Examination Scheme	Credit Scheme
Theory:3 Hours / Week	End Semester Examination: 60 Marks	Theory:03 Credits
Practical: 2 Hours / Week	Continuous Assessment: 40 Marks	TW/PR:01 Credit
	Term work/Practical: 50 Marks	

Course Pre-requisites:

1. Students should have basic knowledge of Computers and Internet.
2. C/C++/Java programming, algorithms & probability.

Course Objectives:

1. Developing the students with the knowledge of advanced computer networking.
2. Developing students with practical knowledge of latest networking technologies.

Course Outcomes: After completion of course, students will able to:

1. Recite the basics of Computer network.
2. Relate detailed structure of data link layer with other layers.
3. Enumerate the concept of Medium Access Control layer.
4. Recite the details of Network layer.
5. Discuss the details of Transport layer.
6. Infer the functionality of Application layer.

Unit I **06 Hours**

INTRODUCTION:

Internet working, Use of the computer network, Physical layer, Networking hardware devices, networking software, types of networks, OSI model working, TCP/IP reference model, Wireless networks, Buffering, Switching, packets, frames, introduction to latest internet technologies.

Unit II **06 Hours**

Data Link Layer:

Data layer working, Framing, Error Control, Flow Control, error detection and correction, data link protocols, IEEE standards.

Unit III **06 Hours**

MEDIUM ACCESS CONTROL SUBLAYER:

MAC layer working, Ethernet, Static & Dynamic Channel Allocation, multiple access protocols, wireless technologies, RFID.

Unit IV **06 Hours**

NETWORK LAYER:

Network layer working, Network layer design issues, routing algorithms, congestion control algorithms, quality of service, IP Addresses, Subnets, Configuring network settings, Firewalls, IP V6, Mobile IP V6.

Unit V **06 Hours**

TRANSPORT LAYER:

Transport layer working, services of transport layer and elements of transport protocols, congestion control in transport layer, Transport protocols-TCP &UDP, Performance issues, Mobile TCP.

Unit VI **06 Hours**

APPLICATION LAYER:

Application layer working, DNS, Email, WWW, Audio & video streaming, Content delivery, Caching in Web Browser remote login, Wireless web, browsers, NFS, SNMP, Telnet, FTP, HTTP, WiMAX, WSN.

Term Work:

- Introduction and configuration of networking devices

  Practical Study of PC, router, Switches, hubs, servers, repeaters, Wi-Fi modem and its configurations
- Introduction to “CISCO’s Packet tracer”. Working and study of CISCO’s Packet tracer
- Implementation of Packet switching using “CISCO’s Packet tracer” software.
- Client-Server setup. Connection establishment between client and server. Study of FTP and HTTP through this setup. Study of packet delivery. Simulation of packet delivery.
- Implementation of static routing using “Packet tracer” software.

- Distance vector routing algorithm. Packet switching using static routing algorithm.
- Implementation of dynamic routing using “Packet tracer” software.
- Dijkstra's shortest path algorithm. Packet switching using dynamic routing algorithm.
- WAP in C to implement routing algorithm using Bellman Fords distance vector algorithm.
- Study of network throughput and efficiency using “Wireshark software”
- Packet delivery from one system to other. Observe its throughput by using Wireshark software.
- Data encryption using “GnuPG” Software.
- Encryption of the data using GnuPG software. Encrypt data and deliver it to other system and
- Observing the change in file size using Wireshark
- Experiment related to NS2/NS3 tool.
- Case study of “Networking commands in Linux based operating system”

Assignment:

NOTE:

The Term Work prescribed in the syllabus is continuous assessment by the concerned subject faculty. For internal Assessment of 10 Marks, students have to submit two assignments based on problems of different types of any programming assignment or theory assignment or any case study or quiz or Multiple Choice Questions etc.

The assignments are to be submitted as a hard copy.

Textbooks:

1. James f. Kurose, Keith w. Ross. Computer networking: a top-down approach. 6th ed.: Pearson.
2. Andrew s. Tanenbaum, David j. Wetherall.Computer networks.5th ed.: Pearson.
3. Forouzan. Data comm.& netw.5e. 5th ed.: McGraw-Hill.
4. William Stallings. Data and computer communications. 8th ed.: Pearson.

Syllabus for Unit Tests:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

DATABASE MANAGEMENT SYSTEM

Teaching Scheme	Examination Scheme	Credit Scheme
Theory: 03 Hours / Week	End Semester Examination: 60 Marks	Theory:03 Credits
Practical: 02 Hours / Week	Continuous Assessment: 40 Marks	TW/PR:01 Credit
	TW/Practical: 50 Marks	

Course Pre-requisites:

1. Discrete mathematics and Data structures.
2. Programming languages.

Course Objectives:

1. To provide a strong formal foundation in database concepts, technology and practice.
2. To design and develop a database schema.
3. To understand Structured query language (SQL), indexing and query processing.

Course Outcomes: After completion of course, students will able to:

1. Differentiate significance of Database Management System over the file processing system.
2. Illustrate the fundamentals of data models and to conceptualize and depict a database system using data models.
3. Analyze and practice Relational Data Model.
4. Apply SQL queries for database definition and database manipulation.
5. Illustrate transaction management concepts like serializability, concurrency control and recovery system.
6. Investigate the knowledge about emerging trends in the area of database for unstructured data and applications for it

Unit I Overview of Database Management System 06 Hours

Overview of DBMS: Data and Information, Database, characteristics of databases, Data Governance and Importance of database, Database Management System (DBMS), structure of DBMS, Objectives of DBMS: Data

Availability, Data Integrity, Data Security, and Data Independence. Three level DBMS architecture and Data Abstraction. Classification of DBMS, File based System; Drawbacks of File based System, Advantages of DBMS.

Database Architecture: Two Tier, Three Tier, Multi-Tier. The DBMS Life Cycle, Information Life Cycle, Roles in Database Environment, Database users and Administrator, Introduction to Data models, Need for abstraction, Situation where DBMS is not necessary, DBMS vendors and their products.

Unit II **06 Hours**

Data modeling and Design

Data modeling: Benefits of Data Modeling, Types of Data Modeling, Phases of Data Modeling, Building Blocks of Entity Relationship(E-R) model, Mapping Constraints, Keys, Weak and strong Entity Sets, Extended-R features, Reduction to relational schemas, a case study on building an ER Model.

Database design: Objectives of Database design, Database Design Tool, Features of Design Tool, merits and De-merits of database design tool, Different anomalies in designing a Database, Functional Dependency, Inference Rules, Closure of set

Unit III **06 Hours**

Relational Model and Algebra

Relational Data Model: CODD's Rule, Structure Part, Integrity Part, Manipulative, Table & Relation. Concept of Key: Super Key, Candidate Key, Foreign Key. Relational Integrity: Entity Integrity, NULL Integrity, Domain Integrity constraint, Referential Integrity, Data structure, Mapping ER model to Relational model.

Relational Algebra: Unary and Binary Operations, Rename Operation, Union Operation, Intersection, Difference, Division, Cartesian Product, Join Operations. Advantage and Limitation of Relational Algebra.

Unit IV **06 Hours**

SQL and Introduction to PL/SQL

SQL: SQL fundamentals, Data Definition Language(DDL), Data Manipulation Language(DML) and Data Control Language(DCL), Basic structure of SQL queries, set operations, Aggregate Functions, Null Values, Domain Constraints, Referential Integrity Constraints, Sub queries, joins, Nested Sub queries, Complex queries, views: Creating, Dropping, Updation using Views.

PL/SQL: Introduction to PL/SQL block, Concept of Cursors, Stored Procedures, Triggers, Java Database Connectivity (JDBC), Open Database Connectivity (ODBC).

Unit V **08 Hours**

Transaction Processing and Query Optimization

Transaction processing: Concept of Transaction and its Management, concurrency Anomalies, Schedules, schedules and recoverability, Serializability, Hierarchy of serializable Schedules, Concurrency control and enforcing serializability, Deadlocks, Lock Granularity, Lock based Concurrency control, Multiple Granularity and Intension Locking, Time stamping Control, Optimistic Control, Evaluation of Concurrency Control Mechanism.

Query Processing and Optimization: Issues in Query Processing, Steps in Query Processing, Query decomposition, Query Optimization: Heuristic query optimization, Transformation rules, Heuristics optimization algorithm. Cost estimation in query optimization: Cost of components of query execution, cost for select and join operation, Query execution Plan.

Unit VI **08 Hours**

Data Warehousing and Data mining

Data Warehousing: Introduction, Evolution of Data Warehouse, Characteristics, and Benefits, Limitation of Data Warehousing, Main Components of Data Warehouse, Conceptual Models, Data Mart, and Online Analytical Processing (OLAP), Teradata Relational Database Management System(RDBMS), Teradata Technology.

Data Mining: Data Mining Concepts, Knowledge Discovery, Goals of Data Mining, Data Mining techniques, Machine learning using WEKA tool.

Emerging Database Technologies: Introduction to unstructured data, NOSQL,

spatial and geographic databases, multimedia databases, Massive Datasets and Hadoop.

<u>Term Work:</u>

The sample practical assignments are given below. This can be used as a guideline and course coordinator can recommend the list of practical assignments.

1. Write a simple PL/SQL program to check whether the given number is palindrome or not and insert into a table reverse if the given number is palindrome?
2. To import various schemas into database system by running the scripts. Schemas are human resource, Order entry, Product Media, Queued shipping, Sales history.
3. To apply the select statements for the given queries.

a. Display employees those who make more than $900 or if their names fall between QUENTIN and ZYRYAB alphabetically.

b. Display employees those who make more than $900 and if their names fall between QUENTIN and ZYRYAB alphabetically.

c. Display the name and job title of all employees who do not have a manager.

d. Display the name, salary and commission for all employees who earn commissions. Sort data in descending order of salary and commissions.

4. To apply the single column functions:
    a. WAQ that displays the employee's names with the first letter capitalized and all other letters lowercase and the length of the names, for all employees whose name starts with J, A, or M. Give each column an appropriate label. Sort the results by the employee's name.
    b. For each employee, display the employee's name and calculate the number of months between today and the date employee was hired. Label the column months worked. Order your results by the number of months employed. Round the number of months up to the closest whole number.
    c. WAQ that produces following for each employee.
    i. <Employee name> earns <salary> monthly but wants < 3 times salary>. Label the column dream salaries
5. To discuss normalization and build normalized schema of Hospital Management system.
6. To demonstrate queries on Joins.
    a. Display the employee name and employee number along with their manager's name and manager number.
    b. To display all employees including king, who as no manager. Order the results by employee number.
    c. Display employee name, department name and all the employee who work in the same department as a given employee.
    d. Create a query to display the name and hire date of any employee hired after employees Davies.

7. To demonstrate queries on aggregate functions
   a. Determine the number of managers without listing them.
   b. Display the Manager number and the salary of the lowest paid employee for that manager.
   c. Display each department name, location, number of employee and the average salary for all employee in that department.
   d. Create a query that displays total number of employees and of that total, the number of employees hired in 1995, 1996, 1997 and 1998.
8. Use WEKA tool to derive analytical model for the given dataset.
9. Apply the association rules on the dataset and derive the inferences from the results given by WEKA tool.
10. Case study on NOSQL database: MongoDB.

<u>Assignment:</u>

NOTE:

The Term Work prescribed in the syllabus is continuous assessment by the concerned subject faculty.

For internal Assessment of 10 Marks, students have to submit two assignments based on problems of different types of any programming assignment or theory assignment or any case study or quiz or Multiple Choice Questions etc.

The assignments are to be submitted as a hard copy.

<u>Textbooks:</u>

1) Silberschatz A., Korth H., Sudarshan S, "Database System Concepts", 6th Edition, McGraw Hill Publishers.
2) Ramakrishna R., Gehrke J., "Database Management Systems", 3rd Edition, McGraw-Hill.

<u>Reference Books:</u>

1) Elmasri R., Navathe S., "Fundamentals of Database Systems", 5th Edition, Pearson Education.
2) Ryan K. Stephens, Ronald R. Plew,"SQL", 4th Edition, Pearson Education.

<u>Syllabus for Unit Tests:</u>

Unit Test -1 Unit – I, Unit – II, Unit - III

Unit Test -2Unit – IV, Unit – V, Unit – VI

ELECTIVE – I: MULTIMEDIA AND MOBILE APPLICATIONS

Teaching Scheme	Examination Scheme	Credit Scheme
Theory: 03 Hours / Week	End Semester Exam: 60 Marks	Theory: 03 Credits
	Continuous Assessment: 40 Marks	

The Students should have

1. Basic knowledge of JAVA Programming.
2. Basic knowledge of Data Communication.

Course Objectives:

1. Create awareness among the students about the basic fundamentals in the design and development of multimedia systems and applications.
2. To make students aware of various techniques used for the development of distributed applications for mobile devices as well as introduce them to the design principles for applications for small devices

Course Outcomes: After completion of course, students will able to:

1. Recite basic building blocks of Multimedia systems and applications.
2. Differentiate between various Compression and Decompression Techniques.
3. Comprehend various Multimedia I/O, Storage and Retrieval technologies.
4. Analyze various issues pertaining to Multimedia Application Design.
5. Report on Multimedia Networks.
6. Discuss on Mobile Application Development.

Unit I **06 Hours**

Introduction to Multimedia:

Multimedia- Building Blocks, Overview of Multimedia Applications, Multimedia Systems- Components, Characteristics, key issues and Challenges, Evolving Technologies for Multimedia Systems, Multimedia Streaming Protocols (MSP).

Multimedia Data: Text and static data, Graphics, Images, audio, video. Multimedia Data Compression- Lossy and Lossless compression techniques.

Unit II **06 Hours**

Compression and Decompression Techniques

Types of Compression, Binary Image Compression Schemes, Color, grayscale, still- video image compression, Discrete Cosine Transform, Video image compression, MPEG Coding methodology, Audio Compression, Data and File format standards- PDF, RTF, TIFF, RIFF, MIDI, JPEG, AVI, JPEG, MPEG.

Unit III **06 Hours**

Multimedia I/O, Audio and Video

Key Technology Issues, Pen Input, Video and Image Display Systems, Print Output Technologies, Image Scanners, Digital Voice and Audio, Video Images and Animation, Full Motion Video, Magnetic Media Technology, WORM optical drives, Cloud-Based Multimedia Storage systems.

Multimedia Databases: Design and Architecture of a Multimedia Database, Organizing Multimedia Databased on The Principle of Uniformity, Media Abstractions, Query Languages for Retrieving Multimedia Data.

Unit IV **06 Hours**

Multimedia Application Design

Types of Multimedia systems - Virtual Reality Design - Components of Multimedia

system - Distributed Application Design Issues - Multimedia Authoring and User Interface - Hypermedia Messaging- Distributed Multimedia Systems, Multimedia Authoring Tools.

Unit V **06 Hours**

Multimedia Networks

Basics of Multimedia Networks, Multimedia Network Communications.

Applications: Quality of Multimedia Data Transmission, Multimedia over IP, Multimedia over ATM Networks, Transport ofMPEG-4, Media-on-Demand(MOD).

Video Broadcasting Standards- HDTV, 4K TV and Ultra HD, Multimedia Content Management (MCM).

Unit VI **06 Hours**

Introduction: Understanding Mobile Platforms, Android as Competition to itself, Building an App in Android, Debugging Android Apps. Building block of Mobile apps: App user Interface Designing, Layout, User Interface elements, Activity states and lifecycle, Mobile Databases such as SQLite and enterprise data access, Windows Mobile OS (Operating System).

Assignments:

NOTE:

The Term Work prescribed in the syllabus is continuous assessment by the concerned subject faculty.

In case of assignments for internal 10 marks students will be assigned two assignments containing problems of different types or any programming assignment and guided for the solutions of the problem.

The assignments are to be submitted as hard copy.

Textbooks

1. Ralph Steinmetz, Klara Nahrstedt –Multimedia computing, Communication & Application– Pearson Education, 6th Edition.
2. Ze-Nian Le, Mark.S. Drew, Jiang Chuan Liu, Fundamentals of Multimedia, Springer, 2nd edition.
3. JakobIversen, Michael Eierman, Learning Mobile App Development- A Hands-On Guide to Building Apps with iOS and Android, Addison-wesley,1st edition, 2013.

Reference Books

1. Tay Vaughan, "Multimedia making It work", Tata Mc Graw Hill 5th Edition 2001.
2. Fred Halsall, "Multimedia Communications", Addison Wesley, 1st Edition, 2000.
3. Andleigh P K and Thakrar K," Multimedia Systems", Addison Wesley , 1999.
4. Jeff Mc Wherter, Scott Gowell, "Professional Mobile Application Development", Wiley Publications.

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT – III

Unit Test -2 UNIT – IV, UNIT – V, UNIT – VI

ELECTIVE – I: SCRIPTING LANGUAGES

Teaching Scheme	Examination Scheme	Credit Scheme
Theory: 3 Hours / Week	End Semester Examination: 60 Marks	Theory:03 Credits
	Continuous Assessment: 40 Marks	

Course Pre-requisites:

1. The course prerequisites are some general familiarity with programming language constructs such as loops, functions, and conditionals. And revision of regular expressions.

Course Objectives:

1. The study of the principles of scripting languages
2. Understand the difference between client side and server side scripting.
3. Understand the basic techniques used to create scripts for automating system administration tasks.

Course Outcomes: After completion of course, students will able to:

1. To Exemplify the fundamentals of scripting languages.
2. To design and implement JavaScript programs using web browser.
3. To analyze the DOM model and JSON.
4. To implement JQuery.
5. To demonstrate basics of PHP and implement the program using web server.
6. To exhibit the fundamentals of python script and implement it using python interpreter.

Unit I **06 Hours**

Introduction to Scripting Languages: Motivation for and applications of Scripting languages; Difference between Scripting languages and non-Scripting languages; Merits of scripting languages; Types of Scripting languages, Client Side Scripting, Scripting for content structuring, Form design, Client Side Validation, dynamic page generation, adding interactivity, Styles, using HTML, DHTML, XHTML, CSS, Java Script. Server Side Scripting, Types of servers - Configuring and Using Web servers, Important features and Examples of other Scripting Languages.

Unit II **06 Hours**

Java Script fundamentals& Working with Java Scripts:

Introduction to Java Script, History, Variables, Data Conversions, Interaction with the User, Operators, Math and Date Objects, Flow control, String Objects, String Methods, Number Objects, Formatting Numbers, Common Array Methods, Associative Arrays.

Implementing Functions: Declaration & Invocation, Passing Arguments, Returning Values, Anonymous Functions and Event Handling.

Unit III **06 Hours**

JavaScript Objects and JSON:

Browser Object Model, Document Object Model (DOM), JSON, jQuery Selectors and Filters, DOM Manipulation, jQuery Events, jQuery Event Model, Common jQuery Events, Delegated and Direct Events. jQuery Plugins, Ajax Requests Using jQuery.

Unit IV **06 Hours**

XML: Creating Markup with XML, Document Type Definition, Schemas Document Object Model, Simple API for XML, Extensible Stylesheet languages, Formatting Objects, Xpath, XLink and Pointer, Introduction to SOAP, Case Studies, Custom markup languages. Parsers: DOM and SAX - Evolution of AJAX.

Unit V **06 Hours**

Python Script: Introduction, Conditional Statements, Looping, Control Statements, String Manipulation, Lists, Tuple, Dictionaries, Functions, Modules, Input-Output, Exception Handling.

Unit VI **06 Hours**

Python Script with OOP: OOPs concepts, Classes in Python, Principles of Object Orientation, Instance Methods, File Organization, Special Methods, Class Variables, Inheritance, Polymorphism, Type Identification, Regular expressions, CGI, Database, Networking, Multithreading, GUI Programming.

Assignment:

Note:The Term Work prescribed in the syllabus is continuous assessment by the concerned subject faculty. For internal Assessment of 10 Marks, students have to submit two assignments based on problems of different types of programming assignment or theory assignment or any case study or quiz or Multiple Choice Questions etc.

The assignments are to be submitted as a hard copy.

Textbooks:

1) Python: The Complete Reference by Martin C. Brown
2) JavaScript: The Complete Reference by Thomas Powell, Fritz Schneider.

Reference Books:

1) JavaScript in 24 Hours, 4th ed. Michael Moncur. 2007. Sams Publishing.
2) Python Tutorial by Guido van Rossum, and Fred Drake, Jr., editor, Release 2.6.4.
3) Programming Python, by Mark Lutz.O'REILLY
4) Xml: The Complete Reference by Willimson, Tata McGraw-Hill Education
5) Beginning Python: From Novice to Professional (2nd Edition) Author: Magnus Lie Hetland.

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

ELECTIVE – I: SOFTWARE PROJECT MANAGEMENT

Teaching Scheme	Examination Scheme	Credit Scheme
Theory: 03 Hours / Week	End Semester Examination: 60 Marks	Theory:03 Credits
	Continuous Assessment: 40 Marks	

Course Pre-requisites:

1. Understanding of Software Development Life cycle(SDLC)
2. Knowledge of management skills and strategies.

Course Objectives:

1. Articulate similarities and differences between IT projects and other types of projects
2. The ability to come up with a project schedule and assign resources
3. Identify project risks, monitor and track project deadlines.
4. The capability to work in a team environment and be aware of different modes of communications

Course Outcomes: After completion of course, students will able to:

1. Practice the process of project management and its application in delivering successful IT projects.
2. Distinguish between the different types of project and follow the stages needed to negotiate an appropriate scope.
3. Plan and Complete the tasks in time effectively and efficiently.
4. Evaluate a project to develop the scope of work, provide accurate cost estimates and to plan the various activities.
5. Identify the resources required for a project and to produce a work plan and resource schedule.
6. Implement the project plans through managing people communications

Unit I **06 Hours**

An overview of IT Project Management

Introduction, the state of IT project management, context of project management, need of project Management, project goals, project life cycle, information technology project methodology (ITPM), project feasibility, project selection and approval, project contracting, PMBOK.

Unit II **06 Hours**

Project Integration & scope Management

Project management process, project Integration management, the project charter, project Management planning framework, the contents of a project plan, the planning process, project scope definition, Management project scope verification, change control, Work Breakdown Structure (WBS), linear responsibility chart.

Unit III **06 Hours**

Project Time Management:

Developing the project schedule, Management Scheduling Charts, logic diagrams and network (AOA, AON), critical path, calendar scheduling and time based network, PDM network, PERT, CPM, Resource loading, resource leveling, allocating scarce resources to projects.

Unit IV **06 Hours**

Project Quality and Cost Management:

Quality tools and philosophies, Quality management system, IT Project quality plan. Cost estimating, Cost escalation, Management system development cycle, Cost estimating process, Elements of budgets and estimates, Project cost accounting and MIS, Budgeting using cost accounts, Cost schedules and forecasts.

Unit V **06 Hours**

Project Human Resource Management:

Organization and project planning, Resource project team, multidisciplinary teams, project Management environment, project leadership, ethics in projects, multicultural projects, Role of project manager, IT governance.

Unit VI **06 Hours**

Project Human Communication Management:

Monitoring and controlling the project, Communication project plan, Project metric, Project Management control, designing the control system, the plan-monitor control cycle, data collection and reporting, reporting performance and progress.

Assignment:

Note: For internal Assessment of 10 Marks, students have to submit two assignments based on problems of different types of any programming assignment or theory assignment or any case study or quiz or Multiple Choice Questions etc.

The assignments are to be submitted as a hard copy

Textbooks:

1., Hughes, cotterel, RajibMalll Tata McGraw Hill Software Project Management.

2. Edwin Bennatan Software Project management.

Reference Books:

1. S.A. Kelkar Software Project Management
2. Whitten, Bentley and Dittman System Analysis and Design Methods

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT – III

Unit Test -2 UNIT – IV, UNIT – V, UNIT – VI

ELECTIVE – I: COMPUTATIONAL GENOMICS

Teaching Scheme	Examination Scheme	Credit Scheme
Theory: 3 Hours / Week	End Semester Examination: 60 Marks	Theory:03 Credits
	Continuous Assessment: 40 Marks	
	Total: 100 Marks	

Course Pre-requisites:

In order to understand the content, gain knowledge and successfully complete this course, students should have a basic understanding of the computer algorithms, Programming skills and basic mathematics.

Course Objectives:

1. The purpose is to enable students to analyze and Interpret data generated by bioinformatics/genomics technology.
2. Enable student to use statistical concepts to design experiments and analyze high dimensional data.

Course Outcomes: After completion of course, students will able to:

1. Apply knowledge of data structures, algorithms and analysis of algorithms to problems in Computational Genomics
2. Recite and Express Knowledge needed to read and interpret cutting-edge results in computational genomics.
3. Express the strengths and limits of current genomics data analysis methods
4. Prepare to lead new research projects in computational genomics
5. Formulate and/or model a biological problem/system as a computer science problem
6. Investigate the applicability of algorithms & techniques in other domains such as text mining, speech recognition, pattern matching and string searching

Unit I **06 Hours**

Introduction to Genome Computational

Introduction to Challenges in Computational biology: gene finding, Sequence alignment, Database lookup, genome assembly, Regulatory Motif Discovery etc., Introduction to Probability: Probability Distribution, Multiple Random Variables, Random Variables assuming infinite values.

Introduction to Markov & Chebycheff inequalities, Hoeffding's inequality, Monte carlo simulation, Cramer's theorem.

Unit II **06 Hours**

Algorithms for computational genomics

Enumeration Approaches: Exhaustive search, Pruning, greedy algorithms, iterative refinement.

Content based indexing: hashing, database lookup, pre-processing.

Iterative Method: Combining sub problems, memorization, dynamic programming.

Statistical Method: Hypothesis testing, Maximum likelihood, Bayes Law, HMM's.

Machining learningte chnique: Supervised and unsupervised learning, classification.

Unit III **06 Hours**

Hidden Markov Processes

Markov Processes: Markov property and state transition matrix, estimating the state transition matrix, stationary Markov chains: recurrent and transient states, hitting probabilities and mean hitting time, Basic properties: Three different looking models, equivalence between three models. The Viterbi algorithm, Baum- Welch Algorithm.

Unit IV **08 Hours**

BLAST THEORY

Anatomy, Problem formulation, Moment generating function, Finding local Matches, Application of main results, Proofs of main results.

BLAST & Database: W-mer Indexing data, Search algorithm, karlin-altschul statistics, Filtering, Two hit Blast, substitution matrices, pigeonhole principle.

Unit V **06 Hours**

Data Mining

Functional Genomics: Splicing & Alternative Splicing, Microarray based Functional genomics.

Data Transformation: data smoothing by discretization, Normalization and standardization, min-max normalization, z-score standardization, use decimal scaling.

Feature Selection: Filter Approaches, Wrapper Approaches.

Clustering Technique: Distance based clustering and measures, K-means Algorithm, k-modes algorithm, Genetic Distance measure, hierarchical clustering, Graph based clustering.

Unit VI **06 Hours**

Classification in Genome and bioinformatics

Bias variance tradeoff in supervised learning, Linear and Nonlinear classifiers, Model complexity and training data set size, support vector machines, Bayesian Approaches, Bayesian Network, Decision tree: tree pruning.

Applications: Sequence alignment using dynamic programming, Scoring matrices for protein sequences.

Assignments:

Note

For internal Assessment of 10 Marks, students have to submit two assignments based on problems of different types of any programming assignment or theory assignment or any case study or quiz or Multiple Choice Questions etc.

The assignments are to be submitted as a hard copy.

Reference Books

[1] Data Mining for Bioinformatics, Sumeet Dua, Pradeep Chowriappa, CRC Press, 06-Nov-2012

[2] Data Mining in Bioinformatics, Jason T. L. Wang, Mohammed J. Zaki, HannuToivonen, Dennis Shasha

[3] Springer Science & Business Media, 30-Mar-2006

[4] Hidden Markov Processes: Theory and Applications to Biology, M. Vidyanagar Princeton University Press, 24-Aug-2014

[5] Theoretical and Computational Methods in Genome Research, Sandor Suhai, Springer Science & Business Media, 06-Dec-2012

[6] Computational Genome Analysis: An Introduction, Richard C. Deonier, Simon Tavaré, Michael Waterman, Springer Science & Business Media, 13-Aug-2007

Syllabus for Unit Tests:

Unit Test -1 UNIT – I, UNIT – II, UNIT – III

Unit Test -2 UNIT – IV, UNIT – V, UNIT – VI

ENGINEERING MATHEMATICS-IV (OPTIONAL SUBJECT)

Teaching Scheme	Examination Scheme	Credit Scheme
Theory: 04 Hours / Week	End Semester Examination: 60 Marks	04 Credits
Practical: -- Hours / Week	Continuous Assessment: 40 Marks	

Course Pre-requisites:

The Students should have knowledge of

1. Determinants
2. Matrices
3. Differentiation
4. Integration of functions
5. Differential equation

Course Objectives:

The course aims at making the students familiar about the most basic numerical methods and concepts like error estimation helpful in various fields of engineering and can be used to simulate the results of various numerical methods.

Course Outcomes:

The student should be able to

1. Derive appropriate numerical methods to solve algebraic and transcendental equations
2. Evaluate the accuracy of common numerical methods.
3. Develop appropriate numerical methods to solve a difference equation.
4. Be familiar with numerical interpolation and approximation of functions, numerical integration and differentiation.
5. Be familiar with numerical solution of ordinary differential equations.
6. To compute Numerical Solution of Partial Differential Equations.

Unit I **08 Hours**

Numerical solutions of algebraic and transcendental equations:

Bisection method, Regula-Falsi method, Newton-Raphson method, Direct iterative method.

Unit II **08 Hours**

Solution of system of linear algebraic equation

Matrix inversion method, Gauss- elimination Method, Jordan's method, Crout's method. Gauss-Seidel and Gauss Jacobi's iterative method.

Unit III **08 Hours**

Difference equation and Solution of difference equations

Definition of difference equations, formation of difference equation. Solution of Homogeneous and non-homogeneous difference equation with constant and variable coefficients using Boole's operator method and generating functions. Simultaneous difference equation.

Unit IV **08 Hours**

Interpolation and Numerical differentiation and integration

Finite difference operator, Interpolation formula with equal and unequal intervals. Divided differences and central differences. Curve fitting : Method of least squares. Straight line, Second degree, parabola, Exponential curve.

Differentiation using forward, backward and divided difference General quadrature formula, Trapezoidal rule, Simpson's 1/3rd rule, Simpson's 3/8th rule, Weddle's rule.

Unit V **08 Hours**

Numerical solution of I order ordinary differential equation

Solution by Euler's,method Euler' Modified method Taylor's series. Runga-kutta method. Milne's Predictors and Correctors method.

Unit VI **08 Hours**

Numerical Solution of Partial Differential Equations

Classification of second order partial differential equations, Solution of Laplace's, Poisson's, heat and wave equations by finite difference methods, Use of method of characteristics for solution of initial and boundary value problems.

Textbooks:

1. Gupta P.P.& Malik G.S., Calculus of Finite Differences and Numerical Analysis, Krishna Prakashan Mandir,Meerut, 21/e, 2006.
2. B.S.Grewal, Engineering Mathematics, Khanna Publishers, 12/e, 2006.

Reference Books:

1. Francis J. Scheid, Schaum's Outline of Numerical Analysis, McGraw-Hill, New York, 1989.
2. S. S. Sastry, Engineering Mathematics, Vol I, II Prentice Hall Publication, 3/e, 2004.
3. C.Ray Wylie & Louis C. Barretle, Advanced Engineering Mathematics, Tata McGraw Hill Publishing Co Ltd., 6/e,2003.

Syllabus for Unit Tests:

Unit Test -1 UNIT – I,II,III

Unit Test -2 UNIT – IV,V,VI

Programming Lab-III

Teaching Scheme	Examination Scheme	Credit Scheme
Practical: 04 Hours / Week	Term Work/Practical: 50 Marks	TW/PR: 02 Credits

Course Pre-requisites:

The Students should have

1. Basic knowledge of object oriented programming.

Course Objectives:

1. To develop ability to use the computational languages necessary for engineering practice.

Course Outcomes: After completion of course, students will able to:

On completion of the course, students will have the ability to:

1. Recite and Express fundamentals of Visual Basic .NET Programming.
2. Illustrate Object Oriented Programming in VB.NET.
3. Demonstrate Windows Forms Application.
4. Apply the concept of developing Console Application.
5. Illustrate Inheritance and Polymorphism.
6. Experiment the data access using ADO.net.

Unit I **06 Hours**

Getting Started with Visual Basic .NET:

Introduction to Visual Basic .NET, MS. Net Project, MSIL, JIT, .NET frame work class library.

Module and Variable: Module and Namespace, Variables, array and structure.

Control flow and Error Handlings: Execution flow control. commands, functions & constants. Error Handling, Debugging .NET applications.

Unit II **06 Hours**

Object Oriented Programming using VB.NET: Methods, properties, constructors, object lifetime, events, Inheritance, in VB.NET., Interfaces, Attributes. Programming in .NET Framework: Array, Lists and collection file, directories and streams.

Unit III **06 Hours**

Windows Forms Application:

Windows forms Applications, Controls, Data Access in visual basic .NET, ADO.NET, Database connectivity.

Unit IV **06 Hours**

C#.net Language Basics

Datatypes, Common Type System, Reference Type and Value Type, Variables Declaration, Concept of Class and Object.

Implicit and Explicit Casting, casting between other datatypes, Boxing and Unboxing, Enum and Constant, Operators, Control Statements, working with Arrays, working with Methods - Pass by value, Pass by reference.

Developing Console Application

Introduction to Project and Solution in Studio, compiling a C# program, Compiling and Building Projects, Using Command Line Arguments, Importance of Exit code of an application.

Unit V **06 Hours**

Inheritance, Polymorphism and Interface in C#.NET

Exception Handling

Exception, Rules for Handling Exception, Exception classes and its important properties, use of try and catch, throwing exceptions, Importance of finally block.

Introduction to Operator Overloading, DLL, DLL Vs Exe., Types of DLL, concept and use of Class Library, Namespace.

WinForms

Introduction to Windows Forms, Controls, Menus and Context Menus, MenuStrip, ToolbarStrip, Graphics and GDI, SDI and MDI Applications, Dialogbox (Modal and Modeless), Form Inheritance.

Unit VI **06 Hours**

Data Access using ADO.NET

Introduction to SQL. Creating Database using VS.NET, Establishing Connection with Database. Executing simple Insert, Update and Delete Statements, Executing Select Statement and using SqlDataReader

Data Access using ADO. NET – Dataset, Advantages of Dataset, Concept of Data Adapter, Data Table, DataGridView, Data Row, Adding / Editing / Deleting rows in the Data Table, Working with Data View.

Multithreading: Creating and Managing Threads, Threads Priority, Thread States, Thread Synchronization & Inter-thread Communication., Using Monitor

Delegates & Events: Delegate Declaration, Sample Application.

Term Work:

Concern Staff should frame assignments on Each Unit.

Textbooks:

Jesse Liberty Learning Visual Basic .NET, O'Reilly.

Andrew Troelsen Pro C# 2008 and the .NET 3.5 Platform, Apress

Reference Books:

Steven Holzner,VB.NET Programming Black Book, Dreamtec Publications.

Matt Telles, C#Programming Black Book, Dreamtec Publications.

Syllabus for Unit Test: NA

OPERATING SYSTEM

Teaching Scheme	Examination Scheme	Credit Scheme
Theory: 3 Hours / Week	End Semester Examination: 60 Marks	Theory: 03 Credits
Practical: 2 Hours / Week	Continuous Assessment: 40 Marks	TW/OR:01 Credit
	TW/Oral: 50 Marks	

Course Pre-requisites:

1. Concept of system software, application software, knowledge of input output devices and its usage

Course Objectives:

1. To enable the students to understand basic concepts of operating system.
2. To brief the students about various design aspects of operating system functionality
3. To give hands on exposure to Linux commands and system.

Course Outcomes: After completion of course, students will able to:

1. Recite and Express theoretical and practical aspects of operating system.
2. Infer the concept of process, thread and Inter process communication
3. Outline the concept of deadlocks, necessary conditions for deadlock and various techniques to handle deadlock
4. Analyze memory management policies.
5. Describe file system File and Input /output structure.
6. Infer LINUX and Android Operating system.

Unit I **06 Hours**

OPERATING SYSTEMS OVERVIEW Computer System Overview-Basic Elements, Instruction Execution, Interrupts, Multiprocessor and Multicore Organization. Operating system Overview-Kernel, Shell, objectives and functions, Evolution of Operating System- Computer System Organization-Operating System Structure and Operations- System Calls, OS Generation and System Boot., Virtual Machines.

Unit II **06 Hours**

PROCESS MANAGEMENT

Processes-Process Concept, Process Scheduling, Operations on Processes, Inter- process Communication; Threads- Overview, Multicore Programming, Multithreading Models; Thread and SMP Management. Process Synchronization - Critical Section Problem, Mutex Locks, Semaphores, Monitors.

Unit III **06 Hours**

MEMORY MANAGEMENT

Main Memory-Contiguous Memory Allocation, Segmentation, Paging, 32 and 64-bit architecture Examples; Virtual Memory- Demand Paging, Page Replacement, Allocation, Thrashing; Allocating Kernel Memory, OS Examples.

Unit IV **06 Hours**

INPUT/OUTPUT SYSTEMS

Mass Storage Structure- Overview, Disk Scheduling and Management; File System Storage-File Concepts, Directory and Disk Structure, Sharing and Protection; File System Implementation- File System Structure, Directory Structure, Allocation Methods, Free Space Management, I/O Systems.

Unit V **08 Hours**

CASE STUDY

Basic Concepts, of LINUX Multifunction Server, Virtualization- Xen, VMware with Linux Host, Android operating system –Features, characteristics, Basic building blocks, Architecture, System services.

Term Work:

The sample practical assignments are given below. This can be used as a guideline and course coordinator can recommend the list of practical assignments.

1. Demonstrate the process creation and various states of a process
2. Apply various scheduling algorithms on a process.
3. Apply Banker’s algorithm
4. Predict whether a system is in a Safe or Unsafe state.
5. Demonstrate various contiguous memory allocation strategies
6. Demonstrate various page replacement strategies
7. Apply disk Scheduling algorithms

Assignments:

Note:

The Term Work prescribed in the syllabus is continuous assessment by the concerned subject faculty.

For internal Assessment of 10 Marks, students have to submit two assignments based on problems of different types of any programming assignment or theory assignment or any case study or quiz or Multiple Choice Questions etc.

The assignments are to be submitted as a hard copy.

Reference Books:

1) Dhananjay M Dhamdhere, 'Operating Systems - A Concept Based approach ', Tata McGraw, Hill publication

2 Abraham Silberschatz, Peter B. Galvin &Grege Gagne (Wiley))'. Operating System Concepts '

3) Sumitabha Das, 'Unix Concepts and Applications, Tata McGraw Hill

4) Milan Milinkovic, 'Operating System: Concepts and Design, Tata McGraw Hill

5) Achyut S. Godbole, 'Operating System with case studies in Unix, Netware and Windows NT' Tata McGraw Hill

6) Karim Yoghmour 'Embedded Android', O'Reilly Publication

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT – VI

Programme: B. Tech.Computer Engineering

DESIGN AND ANALYSIS OF ALGORITHMS

Teaching Scheme	Examination Scheme	Credit Scheme
Theory:3 Hours / Week	End Semester Examination: 60 Marks	Theory: 03 Credits
Practical: 2 Hours / Week	Continuous Assessment: 40 Marks	TW/OR: 01Credit
	TW/Oral: 50 Marks	

Course Pre-requisites:

1. The students should possess the knowledge of Data Structures

Course Objectives:

1. To be able to understand the performance analysis of algorithms.
2. To analyze Dynamic Programming Algorithms.
3. Understand NP-Hard, NP- complete Problems.

Course Outcomes: After completion of course, students will able to:

1. Analyze time complexity
2. Analyze space complexity
3. Discuss Divide and Conquer Method
4. Design algorithms using greedy Methods
5. Infer Backtracking
6. Outline NP-Hard and NP-Complete Problems

Unit I **06 Hours**

Introduction: Algorithm Specification, Pseudocode Conventions, Recursive Algorithms, Performance Analysis: Space Complexity, Time Complexity, Asymptotic Notations, Performance Measurements.

Unit II **06 Hours**

Basic Algorithms: Basics of Probability Theory, Primality testing, Its Advantages and Disadvantages. Elementary Data Structures: Stack, Queues, Binary Trees, Binary Search trees, Heaps, Heap sort, Graphs, Basic Traversal and Search Techniques. Analysis for complexity of all algorithms.

Unit III **06 Hours**

Divide and Conquer: General Method, Binary Search, Finding the maximum and minimum, Merge sort, Quick sort, Performance Measurement, Worst case Analysis. Strassens's matrix multiplication.

Unit IV **06 Hours**

Greedy Method:

The General Method, Knapsack problem, tree vertex splitting, Job sequencing.

Minimum Cost Spanning Trees: Prim's Algorithm, Kruskal's Algorithm, Optimal Merge Patterns, Single-Source shortest path. Dynamic Programming: The General Method, Multistage Graph, All pair shortest path, Single Source shortest path, General Weights, Optimal Binary Search Trees,

String Editing, 0/1 Knapsack, Traveling Salesman Problem.

Unit V **06 Hours**

Backtracking: The General Method, The 8- Queens Problem, Sum of Subsets, Graph Coloring, and Hamiltonian Cycles. Branch and Bound: Least Cost (LC) Search, the 15-puzzle Control abstraction of LC Search, Bounding, FIFO Branch and Bound, LC Branch and Bound.

Unit VI **06 Hours**

Algorithm Complexities: Non deterministic Algorithms, The classes NP- Hard and NP- Complete, Cook's Theorem, NP-Hard Graph Problems, NP- Hard Scheduling Problems, NP-Hard Code Generation Problems. Approximation Problems.

Term Work:

The sample practical assignments are given below. This can be used as a guideline and course

coordinator can recommend the list of practical assignments.

1. Calculate the time complexity of various algorithms.
2. Calculate the space complexity of various algorithms.
3. Implement Knapsack Algorithm.
4. Implement Prim's Algorithm
5. Study and analysis of 8-Queens Problem.
7. Implement Optimal Binary Search Tree.
8. Analyze Quick Sort for Best Case, Worst Case.

Assignments:

Note:

The Term Work prescribed in the syllabus is continuous assessment by the concerned subject faculty. For internal Assessment of 10 Marks, students have to submit two assignments based on problems of different types of any programming assignment or theory assignment or any case study or quiz or Multiple Choice Questions etc.

The assignments are to be submitted as a hard copy.

Text books

1) ElitzHorowith and SartajSahani, S. Rajasekaran, "Fundamentals of Computer Algorithms", Galgotia Publications.
2) Alfred Aho, John E. Hopcroft, "Design and Analysis of Computer Algorithms", Pearson Education

References:

1) Thomas Cormen, Charles E Leiserson, Ronald Rivest, "Introduction to Algorithms, Tata Mc-Graw Hill Publication, Second Edition.
2) Rod Stephens, "Essential Algorithms: A Practical Approach to Computer Algorithms", John Wiley and Sons Publications
3) Jon Kleinberg, Eva Tardos, "Algorithm Design", Pearson Education
4) Robert Sedgewick, Philippe Flajolet, "An Introduction to the Analysis of Algorithms", Adison- Wesley Publication, Second Edition
5) Steven S. Skiena, "The Algorithm Design Manual", Springer Publication, Second Edition.

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT – III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

DIGITAL SIGNAL PROCESSING

Teaching Scheme	Examination Scheme	Credit Scheme
Theory:3 Hours / Week	End Semester Examination: 60 Marks	Theory: 03 Credits
Practical: 2 Hours / Week	Continuous Assessment: 40 Marks	TW/OR: 01 Credit
	TW/Oral: 50 Marks	

Course Pre-requisites:

Basic knowledge of Mathematics, Data communication and Microprocessor

Course Objectives:

1. To analyze discrete time signals and systems in time and frequency domain.
2. To represent structures for discrete time systems.
3. To design digital filters and implement using open source software
4. To understand DSP processors and DSP based applications in real world.

Course Outcomes: After completion of course, students will able to:

1. Classify discrete time signals and systems
2. Analyze LTI system in frequency domain using FT and DFT
3. Analyze discrete time signals and LTI system using ZT.
4. Design structures for discrete time systems.
5. Design and Implement FIR and IIR filters.
6. Examine Architecture and features of DSP processor and describe applications of DSP

Unit I **06 Hours**

Signals and Systems:

Basic elements of DSP system, Analog to Digital conversion process, Aliasing effect Standard signals, Discrete time systems, classification and properties of discrete time systems, Linear Shift Invariant (LSI) systems, Impulse response, Conditions for causality and stability for LTI systems, Linear convolution.

Unit II **06 Hours**

Fourier Transform

Fourier transform of standard signals, Discrete Fourier Transform (DFT), DFT of standard signals, properties of DFT, Circular convolution, correlation, Radix-2 FFT algorithms, Decimation in Time (DIT) and Decimation in Frequency (DIF), Inverse DFT and computation of IDFT, relation between ZT, FT and DFT.

Unit III **06 Hours**

Z transform

Region of Convergence (ROC) and its properties, Z transforms of standard signals, properties of Z transform, Inverse Z transform, System functions from Z transform and pole-zero plots, computation of poles and zeros.

Unit IV **06 Hours**

Structures for discrete time systems: Implementation of general difference equation, Basic structures for FIR systems, Basic structures IIR systems, Representation of structures using signal flow graphs, Feedback in IIR systems, Linear phase FIR filters.

Unit V **06 Hours**

Digital Filters

Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) filters, IIR filter design using impulse invariance and bilinear transformation method, Butterworth approximation, FIR filter design using Windows, Gibb's phenomenon.

Unit VI **06 Hours**

DSP Processors Architecture and applications

DSP processors fundamentals, Characteristics, Evolution of DSP processors, Comparison of DSP processor and microprocessor, conventional and Enhanced architecture of DSP

Case study: - TMS 320C64X

Applications of DSP in image processing, Feature extraction and pattern matching. Applications of DSP in speech processing, Speech recognition, speech synthesis, Echo cancellation.

Case study: - Multi-rate signal processing.

Term Work:

The sample practical assignments are given below. This can be used as a guideline and course coordinator can recommend the list of practical assignments.

1. WAP to generate samples of Sine, Cosine, Square and Random signal.
2. WAP to compute linear convolution
3. WAP to find N point DFT of a given sequence.
4. WAP to compute Circular convolution
5. WAP to implement Radix-2 DIT FFT Algorithm.
6. WAP to compute Z transform and draw pole zero plot
7. WAP to compute Z transform and draw pole zero plot
8. Find Fourier transform of various window functions
9. Assignment based on DSP applications in Image processing
10. Assignment based on DSP applications in speech processing

Assignments:

Note:

In case of assignments for internal 10 Marks students will be assigned two assignments based on different computer organization and architecture concepts and guided for the respective assignment.

The assignments are to be submitted as a hard copy

Text books

1. John G. Proakis, D.G.Manolakis, "Digital Signal Processing", Pearson Prentice Hall.
2. B.P. Lathi, "Signal Processing and Linear Systems", Oxford University Press.

References:

1. B.Venkataramani, M. Bhaskar, "Digital Signal Processors", Architecture programming & applications, TMH.
2. S.K. Mitra, "Digital Signal Processing Computer Based Approach", TMH.
3. M.H. Hayes "digital signal Processing" Schaum's outlines TMH
4. "TMS 320 C64X CPU & instruction set" Texas instruments reference guide
5. Fredic Harris "Multirate signal processing for communication System", PHI

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

COMPUTER ORGANIZATION AND ARCHITECTURE

Teaching Scheme	Examination Scheme	Credit Scheme
Theory:3 Hours / Week	End Semester Examination: 60 Marks	Theory: 04 Credits
Tutorial: 1 Hour / Week	Continuous Assessment: 40 Marks	

Course Pre-requisites:

1. Concept of digital logic and microprocessors
2. Basic Understanding of Computer System

Course Objectives:

1. To understand the core concepts about the computer architecture and computer organization.
2. To understand the design of the various functional units of computer system.

Course Outcomes: After completion of course, students will able to:

1. Solve fixed point and floating point arithmetic problems using algorithms.
2. Infer the architecture and functions of Central Processing Unit.
3. Outline the design approaches and functional requirements for implementing control unit.
4. Describe the I/O organization and interconnections
5. Analyze the characteristics of memory system.
6. Infer multiprocessor configuration and modern computer organization.

Unit I **06 Hours**

CPU structure and function:

Components and functions of computer system, CPU architecture, Processor organization, Register Organization, Instruction Cycle, instruction pipeline. RISC and CISC architecture, The Pentium Processor, Power PC., Superscalar processors.

Unit II **06 Hours**

Computer Arithmetic

ALU, Fixed and Floating point numbers, Integer arithmetic, Booth's algorithm, Hardware implementation, Restoring and Non-Restoring Division algorithm, Floating point representation, IEEE standards.

Unit III **06 Hours**

Control Unit Organization

Micro-operation and their Register Transfer Language (RTL) specification, Hardware control design methods and implementation, Micro program control, Micro instruction Sequencing, Micro instruction execution.

Unit IV **06 Hours**

Input/output Organization

Computer system, I/O modules, Programmed I/O, Interrupt driven I/O, Interrupt processing, I/O channels and Processor, DMA, Interface circuits, Bus interconnection, Bus arbitration, Standard buses, Standard interfaces, PCI, SCSI, USB bus.

Unit V **06 Hours**

Memory Organization

Internal memory, Characteristics of memory systems, Memory hierarchy, Error correction, Cache memory organization, Mapping, Replacement algorithms, Pentium cache organization, DDR3 Memory Organization, NUMA and UMA caches.

Hard disk drives, RAID levels, Flash drives, USB stick.

Unit VI **06 Hours**

Multiprocessor Configuration:

Flynn’s classification for multiprocessor system, closely coupled, loosely coupled and tightly coupled configurations. Problems of Bus contentions, Inter Processor communication.

Multi core systems, Virtual processors. Case study: - Processor architecture in latest Mobile /Laptop.

Assignments:

Note:

The Term Work prescribed in the syllabus is continuous assessment by the concerned subject faculty.

For internal Assessment of 10 Marks, students have to submit two assignments based on problems of different types of any programming assignment or theory assignment or any case study or quiz or Multiple Choice Questions etc.

The assignments are to be submitted as a hard copy.

Text Books

1. William Stalling, “Computer Organization and Architecture”, Fifth edition, PHI

2. Zaky S, Hamacher, “Computer Organization”, Fifth Edition, McGraw-Hill Publications, 2001, ISBN 0071122184

Reference books

1. John P Hayes, “Computer Architecture and Organization”, McGraw-Hill Publication, 2001, ISBN 0071004793.
2. Carpinelli, “Computer systems Organization & Architecture”, Pearson Education India, 2001, ISBN: 8177587676, 9788177587678
3. Morris Mano, “Computer System Architecture, Pearson”, Third edition (2008), ISBN: 978-8131700709
4. Andrew S. Tanenbaum “Structured Computer Organization “(5th Edition) ISBN-13: 978-0131485211
5. D.A. Patterson and J.L. Hennessy, "Computer Organization and Design - The Hardware/Software Interface", Morgan KaufmannISBN: 9780124077263

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

ELECTIVE –II: VLSI

Teaching Scheme	Examination Scheme	Credit Scheme
Theory: 03 Hours / Week	End Semester Examination: 60 Marks	Theory:03 Credits
	Continuous Assessment: 40 Marks	

Course Pre-requisites:

1. Analog and Digital Electronics.
2. Semiconductor physics.

Course Objectives:

1. Introduce students to VLSI Design.
2. Introduce students to fabrication and testability techniques.
3. Introduce students to design Low-Power CMOS Logic Circuits.
4. Introduce students to design and simulate digital circuits using FPGA.

Course Outcomes: After completion of course, students will able to:

1. Express concept of Microelectronics and Introduction to MOS Technologies.
2. Outline Layout Design and Tools.
3. Infer Combinational Logical& Sequential Systems.
4. Identify and discuss key problems Dynamic Logic Circuits and its solutions.
5. Relate importance of Low-Power CMOS Logic Circuits.
6. Apply useful criteria for guiding design and evaluation of Chip Input and Output (I/O) Circuits.

Unit I **06 Hours**

Review of Microelectronics and Introduction to MOS Technologies: MOS, CMOS, BiCMOS Technology. Basic Electrical Properties of MOS, CMOS & BiCMOS Circuits: Ids – Vds relationships, Threshold Voltage VT, Gm, Gds and ωo, Pass Transistor, MOS, CMOS & Bi CMOS Inverters, Zpu/Zpd, MOS Transistor circuit model, Latch-up in CMOS circuits.

Unit II **06 Hours**

Layout Design and Tools:

Transistor structures, Wires and Vias, Scalable Design rules, Layout Design tools. Logic Gates & Layouts:

Static Complementary Gates, Switch Logic, Alternative Gate circuits, Low power gates, Resistive and Inductive interconnect delays.

Unit III **06 Hours**

Combinational Logic:

Layouts, Simulation, Delay, Interconnect design, Power optimization, Switch logic and Gates.

Sequential Systems:

Memory cells and Arrays, Clocking disciplines, Design, Power optimization, Design validation and testing.

Unit IV **06 Hours**

Dynamic Logic Circuits:

Introduction, Basic Principles of Pass Transistor Circuits, Voltage Bootstrapping, Synchronous Dynamic, Circuit Techniques, Dynamic CMOS Circuit Techniques, High Performance Dynamic CMOS Circuits, Semiconductor Memories: Introduction, Dynamic Random Access Memory (DRAM), Static Random Access Memory (SRAM), Nonvolatile Memory, Flash Memory, Ferroelectric Random Access Memory (FRAM).

Unit V **06 Hours**

Low-Power CMOS Logic Circuits:

Introduction, Overview of Power Consumption, Low-Power Design Through Voltage Scaling, Estimation and Optimization of Switching Activity, Reduction of Switched Capacitance, Adiabatic Logic Circuits. BiCMOS Logic Circuits: Introduction, Bipolar Junction Transistor (BJT): Structure and Operation, Dynamic Behavior of BJTs, Basic BiCMOS Circuits: Static Behavior, Switching Delay in BiCMOS Logic Circuits, BiCMOS Applications.

Unit VI **06 Hours**

Chip Input and Output (I/O) Circuits:

Introduction, ESD Protection, Input Circuits, Output Circuits and L(di/dt)

Noise, On- Chip Clock, Generation and Distribution, Latch-Up and Its Prevention. Design for Manufacturability: Introduction, Process Variations, Basic Concepts and Definitions, Design of Experiments and Performance Modelling, Parametric Yield Estimation, Parametric Yield Maximization, Worst-Case Analysis, Performance Variability Minimization. ASIC Design Flow, Introduction to Verilog, Language Constructs and Conventions in Verilog, Gate Level Modeling, Architecture of FPGA.

Assignments:

Note:

1. For internal assessment of 10 Marks, students have to submit two assignments based on problems of different types of any programming assignment or theory assignment or any case study or quiz or Multiple Choice Questions etc.
2. The assignments are to be submitted as a hard copy.

Reference books

1) Essentials of VLSI Circuits and Systems, K. Eshraghian Eshraghian. D, A. Pucknell, 2005, PHI.

2) Modern VLSI Design – Wayne Wolf, 3rd Ed., 1997, Pearson Education.

3) Introduction to VLSI Systems: A Logic, Circuit and System Perspective – Ming-BO Lin, CRC Press, 2011.

4) Principals of CMOS VLSI Design – N.H.E Weste, K. Eshraghian, 2nd Ed., Addison Wesley.

5) Sung Mo Kang & YosufLeblebici, “CMOS Digital Integrated Circuits: Analysis and Design”, Tata McGraw- Hill, Third Edition

6) Neil Weste and K. Eshragian, “Principles of CMOS VLSI Design: A System Perspective”, Second Edition, Pearson Education (Asia) Pvt. Ltd. 2000.

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

ELECTIVE –II: NATURAL LANGUAGE PROCESSING

Teaching Scheme	Examination Scheme	Credit Scheme
Theory:3Hours / Week	End Semester Examination: 60 Marks	Theory:03 Credits
	Continuous Assessment: 40 Marks	

Course Pre-requisites:

Basic understanding of Theory of Computer Science, Systems Software, Basic mathematics, Probability theory.

Course Objectives:

1. To understand approaches to syntax and semantics in Natural Language Processing and levels of language analysis.
2. To understand current methods for statistical approaches to machine translation.
3. To develop problem solving abilities using Mathematics.

Course Outcomes: After completion of course, students will able to:

1. Outline the Natural Language Processing basics and language representation.
2. Apply various parsing algorithms and its application.
3. Describe language modeling and various clustering techniques.
4. Recite the use of Machine Translation in Natural Language Processing.
5. Apply various filtering methods and semantic networks.
6. Infer advanced tools and parsers for Natural Language Processing.

Unit I **06 Hours**

Introduction to Natural Language Understanding

The Study of Language Applications of Natural Language Understanding, Evaluating Language Understanding Systems, The Different Levels of Language Analysis, Representations and Understanding the Organization of Natural Language Understanding Systems, Structure of Natural Language Processing (NLP).

Unit II **06 Hours**

Grammars for Natural Language

Parsing Algorithms, Robust and Scalable Parsing on Noisy Text as in Web documents, Hybrid of Rule Based and Probabilistic Parsing, Human Preferences in Parsing Encoding Uncertainty: Shift-Reduce Parsers, Deterministic Parser Techniques for Efficient Encoding of Ambiguity Partial Parsing, Part of speech tagging

Unit III **06 Hours**

Natural Language Processing Modeling

Automatic Morphology Learning, Named Entities, Maximum Entropy Models, Random Fields, Estimation Techniques, and Language Modeling, Parsing and Syntax, The EM Algorithm in Natural Language Processing, Stochastic Tagging, and Log-Linear Models, Probabilistic Similarity Measures and Clustering, Machine Translation, Discourse Processing: Segmentation.

Unit IV **06 Hours**

Natural Language Understanding Methods

Finite State Machine Based Morphology; Automatic Morphology Learning;

Unsupervised Methods in NLP, Introduction to HMM, HMM Ergodic models, Graphical Models for Sequence Labeling in NLP, Probabilistic parsing, Forward Backward probability, Viterbi Algorithm.

Unit V **06 Hours**

Ambiguity Resolution

Selectional Restrictions, Semantic Filtering Using Selectional Restrictions, Semantic Networks, Statistical Word Sense Disambiguation, Statistical Semantic Preferences, Combining Approaches to Disambiguation.

Unit VI **06 Hours**

Advanced tools, techniques and applications of NLP

Sentiment Analysis; Text Entailment; Robust and Scalable Machine Translation; Cross Lingual Information Retrieval, Some applications like machine translation, database interface, Programming language Python Natural Language Tool Kit (NLTK), NLP applications in web mining and text mining, Parsers for NLP such as Stanford, Open NLP.

Assignments:

Note:

1. For internal Assessment of 10 Marks, students have to submit two assignments based on problems of different types of any programming assignment or theory assignment or any case study or quiz or Multiple Choice Questions etc.
2. The assignments are to be submitted as a hard copy.

Text books

1) James Allen, “Natural Language Understanding”, Pearson Publication, ISBN: 978-81-317-0895-8 2nd Edition

2) D. Jurafsky, J. H. Martin, “Speech and Language Processing”, Pearson Education, 2002, 2nd Edition

Reference books

1) Christopher D. Manning, HinrichSchutze, “Foundations of Statistical Natural Language Processing”, 1st edition, Cambridge, Massachusetts, 1999.

2) Tanveer Siddiqui, US Tiwari, “Natural Language Processing and Information Retrieval” Oxford Higher Education.

3) Daniel M. Bikel, ImedZitouni, “Multilingual Natural Language Processing Applications” Pearson Education 1st edition.

4) Lutz and Ascher "Learning Python" O'Reilly ISBN: 0596002815

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

ELECTIVE – II: HUMAN COMPUTER INTERACTION

Teaching Scheme	Examination Scheme	Credit Scheme
Theory: 03 Hours / Week	End Semester Examination: 60 Marks	Theory:03 Credits
	Continuous Assessment: 40 Marks	

Course Pre-requisites:

1. Knowledge of user interface and human psychology.
2. Understanding of Input Output devices.

Course Objectives:

1. To apply knowledge of human psychology, design process on user interface development projects.
2. To provide the user interface designer with concepts and strategies for making design decisions.
3. To expose the user interface designer to tools, techniques and ideas.
4. To identify the importance of good user interface design.

Course Outcomes: After completion of course, students will able to:

1. Express the concept of human computer interaction.
2. Describe the principles of human computer interaction.
3. Outline design goals and standards of HCI designs.
4. Identify and discuss key problems in HCI and its solutions.
5. Recite the importance of software tools and techniques of human factors in developing an interactive system.
6. Apply useful criteria for guiding design and evaluation of user interfaces.

Unit I **06 Hours**

Introduction:

Human interaction with computers, importance of human characteristics, human consideration, Motivations for Human Factors in design, Eight golden rules of HCI, Murphy's law. Human Psychology and understanding. Case study on human factors.

Unit II **06 Hours**

The graphical user interface:

Popularity of graphics, the concept of direct manipulation, graphical system, Web user – Interface popularity, Principles of user interface. Current trends in GUI.

Unit III **06 Hours**

Design process:

Importance of user interface, definition, importance of good design. Benefits of good design, study of design process components, three Pillars of design, Social Impact for early design review.

Unit IV **06 Hours**

Software Tools and Techniques in Interface:

Specification methods, interface building tools, Interaction Devices: Keyboard and function keys, pointing devices, speech recognition digitization and generation, image and video displays, pointing devices, navigation schemes.

Unit V **06 Hours**

Screen Designing:

Design goals, Screen planning and purpose, organizing screen elements, ordering of screen data and content, screen navigation and flow – visually pleasing composition, presentation information simply and meaningfully information retrieval on web, Case study on screen design. Evaluation of user interface.

Unit VI **06 Hours**

Windows &Components:

Selection of window, individual, multiple window design, image browsing and tightly coupled windows. Selection of devices based and screen based controls. Components – text and messages, icons and increases, Multimedia, colors, uses, problems, choosing colors.

Assignments:

Note:

For internal assessment of 10 Marks, students have to submit two assignments based on problems of different types of any programming assignment or theory assignment or any case study or quiz or Multiple Choice Questions etc.

The assignments are to be submitted as a hard copy.

Reference books

1) Prece, Rogers, "Sharps Interaction Design", Wiley India.
2) Ben Shneiderman," Designing the user interface". 3rd Edition, Pearson Education Asia
3) Soren Lauesen, "User Interface Design", Pearson Education.
4) Alan Cooper, Robert Reimann, David Cronin, "Essentials of Interaction Design", Wiley.
5) Alan Dix, Janet Fincay, GreGoryd, Abowd, Russell, Bealg," Human Computer Interaction", Pearson Education.

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

ELECTIVE – II: DATA STORAGE SYSTEMS

Teaching Scheme	Examination Scheme	Credit Scheme
Theory:3 Hours / Week	End Semester Examination: 60 Marks	Theory:03 Credits
	Continuous Assessment: 40 Marks	

Course Pre-requisites:

In order to understand the content, gain knowledge and successfully complete this course, students should have a basic understanding of the computer architecture, file systems, operating system and networking.

Course Objectives:

1. To enlighten the students with the knowledge of storage systems and to gain exposure to the storage industry

Course Outcomes: After completion of course, students will able to:

1. Describe the data storage systems and its elements
2. Evaluate various types of intelligent storage systems
3. Describe and Evaluate the deployment model of storage systems
4. Evaluate the various storage networking technologies
5. Infer the key processes for managing the storage infrastructure
6. Determine the appropriate storage solution for a given scenario

Unit I **06 Hours**

Introduction to Data Storage

Data, Data Variety, Information, knowledge, Big Data, Data explosion, Storage, Storage devices and its types, Memory hierarchy, Secondary storage, tertiary storage. Storage architecture and its evolution, Data center, Components of Data center, Managing Data center.

Introduction to Application workloads, DBMS, Compute, Memory Virtualization, Device Driver, Volume Manager, File System, Object storage, Block storage, Server Virtualization.

Unit II **06 Hours**

File System and Storage System

Local file system, Journaling, Snapshots, Network file systems, Principle, NAS, SAN, DAS.

Case Study: DAFS, Shared Disk File System, GPFS, Comparison of FC SAN, FCoE SAN, iSCSI SAN and NAS. Scale up and scale out architectures.

Protocols: SCSI, iSCSI, SC,FCoE, CIFS vs NFS

Unit III **06 Hours**

I/O path and Storage Virtualization

Define Virtualization, define storage virtualization, I/O path, Physical channel, Elements in Physical I/O Path, software Layers of I/O Path, Software stack, I/O path virtualized, Demerits and need of storage virtualization, Virtualization entity, replaceable storage device, use of dynamic storage allocation, use of data migration, virtualization on block and file level, virtualization at network level, symmetric and asymmetric.

Unit IV **06 Hours**

Function and Network technologies

Functions: Instant Copies, Remote Mirroring, consistency groups, LUN Masking, Availability of Disk Storage systems.

Storage network: Transmission technique and protocol, SCSI and storage network, Fiber channel, Link, ports and Topology, IP storage, InfiniBand.

Unit V **06 Hours**

Performance Monitoring and Management

Performance management: Why analyze performance, Capacity Management: capacity planning, I/O Capacity, SLA's, Reactive Style, Casually Observant, Actively Observant, Proactive style, Performance Lifecycle, Performance Hierarchy. Performance Metrics: IOPS, part of an I/O, Throughput, Latency, Utilization, understanding application workload.

Tools: Linux performance commands: TOP, IOSTAT, SAR, IOMeter and Iozone.

Assignments:

Note:

For internal Assessment of 10 Marks, students have to submit two assignments based on problems of different types of any programming assignment or theory assignment or any case study or quiz or Multiple Choice Questions etc.

The assignments are to be submitted as a hard copy.

Reference books

1) Information Storage and Management, second edition, EMC education Services, John Wiley & Sons, Inc.

2) Storage Networks Explained, second edition, Ulf Troppens, Rainer erkens, Wolafka, haustein, Wolfgang, Wiley, SNIA.

3) Introduction to storage area networks and system networking, An IBM Redbooks Publication, Jon Tate, Pall Beck, Hector Hugo Ibarra, Shanmuganathan Kumaravel, Libor Miklas

4) The Design and Implementation of a Robust-storage-system Architecture, Robert C. Good, University of Waterloo, 1995,

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

PROGRAMMING LAB- IV

Teaching Scheme	Examination Scheme	Credit Scheme
Practical: 4 Hours / Week	Term Work (Practical): 50 Marks	Credits: 02 Credits

Course Pre-requisites:

1. Students are expected to have some rudimentary knowledge of computer networking in general and a working knowledge of system programming in C/C++ under Unix/Linux.

Course Objectives:

1. To understand inter-process and inter-system communication.
2. To understand socket programming in its entirety.
3. To understand usage of TCP/UDP / Raw sockets.

Course Outcomes: After completion of course, students will able to:

1. Outline the fundamentals of Unix environment.
2. State Process handling.
3. Recite Inter-Process Communication
4. Implement Unix Socket Programming.
5. Infer fundamental concepts of Shell Programming.
6. Implementation of shell meta-characters.

Unit I **06 Hours**

Overview of Unix OS

UNIX Architecture, UNIX Standardization, Files and Directories, Unix Installation, Input and Output, Programs and Processes, Error Handling, Logging in, User Identification, System Calls and Library Functions, Signals & Time Values

Unit II **06 Hours**

Process Environment

Process Identifiers, Fork Function, Vfork Function, Exit Functions, Wait and Waitpid Functions, Waitid Function, Wait3 and Wait4 Functions, Process Accounting, User Identification, Terminal Logins, Network Logins, Process Groups, Sessions, Introduction of Daemon Processes. Performance Monitoring Tools: Process Status (ps), Top, Xosview, Treeps.

Unit III **06 Hours**

Inter-process Communication

System V IPC. Message Passing: Pipes and FlFOs, Message Queues: System V, Synchronization: Mutexes and Condition Variables, Read-Write Locks, Record Locking, System V Semaphores, System V Shared memory.

Unit IV **06 Hours**

Network IPC: Sockets

Introduction, Socket Descriptors, Addressing, Connection Establishment, Transport Layer, Socket Introduction, TCP Sockets, UDP Sockets, Raw Sockets, Socket Options, I/O Multiplexing, Name and Address Conversions.

Unit V **06 Hours**

Introduction to Unix Shell Programming

Types of Shells, Interactive Shell Scripts, Shell variables, Shell keywords, Positional Parameters, Command line arguments, shell script Arithmetic, Control Instructions, Loop control structure.

Shell Metacharacters: Filename Substitution Metacharacters, I/O Redirection,

Process execution, Conditional execution using && and ||, Quoting metacharacters, Special Parameters, Debugging a Script $* and S@

Unit VI **06 Hours**

Unix Network Tools

Unix Network Commands related to Connectivity, network interface commands, Routing, Arp, NFS/NIS etc.

Windows power shell: purpose of PowerShell, Windows management framework, cmdlets and understanding syntax.

Term Work:

The sample practical assignments are given below. This can be used as a guideline and course coordinator can recommend the list of practical assignments.

1. Introduction to Unix commands using vi editor.
2. Implementation of Unix System calls
3. Write a program to implement message passing using Pipes and FIFOs.
4. Write a program to implement synchronization using semaphores.

5 Write a program to implement synchronization using Mutex variable.

6 Write a program to implement client-server communication using TCP sockets.

7. Write a program to implement client-server communication using UDP sockets.

8. Write a program to implement client-server communication using Raw sockets.

9. Write a shell script to pass command line arguments.

10. Write a shell script using metacharacters for I/O redirection.

Reference books

1. W. Richard Stevens, Stephen A. Rago, “Advanced Programming in Unix Environment”,3rd Edition, Addison-Wesley Professional.
2. W. Richard Stevens “UNIX Network Programming-Interprocess communication”, Volume 2, Second Edition, Prentice Hall Publication.
3. “UNIX Network Programming: The Sockets Networking Api”, Volume 1, 3rd Edition, Addison- Wesley Professional Computing.

BHARATI VIDYAPEETH
(DEEMED TO BE UNIVERSITY) Pune.

Faculty of Engineering & Technology
B.Tech. (Computer Engineering)
Program Curriculum
(2014 Course)

COURSE STRUCTURE AND SYLLABUS
(Choice Based Credit System - 2014 Course)
B.Tech. (Computer Engineering)
Program Curriculum Sem VII & VIII

VISION OF UNIVERSITY:

Social Transformation through Dynamic Education

MISSION OF UNIVERSITY:

- To make available quality education in different areas of knowledge to the students as per their choice and inclination.
- To offer education to the students in a conducive ambience created by enriched infrastructure and academic facilities in its campuses.
- To bring education within the reach of rural, tribal and girl students by providing them substantive fee concessions and subsidized hostel and mess facilities.
- To make available quality education to the students of rural, tribal and other deprived sections of the population.

VISION OF THE INSTITUTE:

To be World Class Institute for Social Transformation Through Dynamic Education.

MISSION OF THE INSTITUTE:

- To provide quality technical education with advanced equipment, qualified faculty members, infrastructure to meet needs of profession and society.
- To provide an environment conducive to innovation, creativity, research, and entrepreneurial leadership.
- To practice and promote professional ethics, transparency and accountability for social community, economic and environmental conditions.

VISION OF THE DEPARTMENT

To be focused on innovative and quality education in computer science and engineering that prepares professionals for development of society.

MISSION OF THE DEPARTMENT

- To provide academic environment for the development of skilled professionals
- To cultivate research culture that contributes to the sustainable development of the society.
- To enhance academic and industry collaborations for global exposure.

Programme: B. Tech.Computer Engineering

PROGRAM EDUCATIONAL OBJECTIVES

The students of B.TECH. (Computer Science and Engineering), after graduating will able to,

1. Demonstrate technical and professional competencies by applying Engineering Fundamentals, knowledge of computing and technologies.
2. Exhibit effective personality, good communication and team building skills
3. Adopt to the latest trends in the field of computer science and engineering.

PROGRAM SPECIFIC OUTCOMES

1. To design, develop and implement computer programs on hardware towards solving problems.
2. To employ expertise and ethical practice through continuing intellectual growth and adapting to the working environment.

PROGRAM OUTCOMES

a. Apply the knowledge of mathematics, science, engineering, and computing to provide a solution of complex engineering problems.

b. Identify, analyse complex engineering problems to derive conclusions using computer science and engineering knowledge.

c. Outline resolutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration, societal, and environmental considerations.

d. Use existing research knowledge and research techniques including design of experiments, data analysis, and synthesis to provide valid inferences.

e. Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools.

f. Apply inferences obtained by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the subsequent responsibilities relevant to the professional engineering practice.

g. Recognize the impact of the professional engineering solutions in societal and environmental contexts to demonstrate the knowledge for sustainable development.

h. Apply ethical principles and execute professional ethics and responsibilities and norms of the engineering practice.

i. Work effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary surroundings.

j. Talk effectively on complex engineering activities with the engineering community such as being able to comprehend and write effective reports and design documentation, make effective presentations.

k. Prove knowledge and understanding of the engineering and management principles and apply these to one's work, as a member and leader in a team.

l. Recognise the need for and have the preparation and ability to engage in independent and life-long learning in context of technological change.

CORELATION BETWEEN GRADUATE ATTRIBUTES AND PROGRAMME OUTCOMES

Graduate Attributes/ Programme Outcomes	a	b	c	d	e	f	g	h	i	j	k	l
Engineering Knowledge	✓											
Problem Analysis		✓										
Design/Development of solutions			✓									
Conduct Investigations of Complex Problems				✓								
Modern Tool Usage					✓							
The Engineer and Society						✓						
Environment and Sustainability							✓					
Ethics								✓				
Individual and teamwork									✓			
Communication										✓		
Project management and finance											✓	
Life-long learning												✓

Bharati Vidyapeeth University, Pune
Faculty of Engineering & Technology
Programme: B. Tech. (Computer) – SEM VII – 2014 Course

Sr. No.	Subject	Teaching Scheme (Hrs/Week)			Examination Scheme (Marks)							Credits		
		L	P/D	T	End Semester Examination	Continuous Assessment			TW & Practical	TW & Oral	Total	Theory	TW	Total
						Unit Test	Attendance	Assignments						
1	Distributed Systems	3		1	60	20	10	10	--	--	100	3	--	3
2	Network Security and Cryptography	3	2	--	60	20	10	10	50	--	150	3	1	4
3	Big Data Analytics and Architecture	3		--	60	20	10	10	--	--	100	3	--	3
4	Compiler Design	3		--	60	20	10	10	--	--	100	3	--	3
5	Elective III	3		--	60	20	10	10	--	--	100	3	--	3
6	Programming Lab - V	--	2	--	--	--	--	-	50	--	50	--	1	1
7	*Industrial Training	--		--	--	--	--	--	--	50	50	--	3	3
8	*Seminar	--	2		--	--	--	-	--	50	50	-	1	1
9	*Project Stage- I	--	8		--	--	--	-	--	50	50	-	4	4
	TOTAL	**15**	**14**	**1**	**300**	**100**	**50**	**50**	**100**	**150**	**750**	**15**	**10**	**25**

Elective -III: a) Grid Computing b) VLSI and Embedded System c) Artificial Intelligence and Robotics d) Ad-Hoc and Sensor Networks

Bharati Vidyapeeth University, Pune
Faculty of Engineering & Technology
Programme: B. Tech. (Computer) – SEM VIII– 2014 Course

Sr.no	Subject	Teaching Scheme (Hrs/Week)			Examination Scheme (Marks)							Credits		
		L	P/D	T	End Semester Examination	Continuous Assessment			TW & Practical	TW & Oral	Total	Theory	TW	Total
						Unit Test	Attendance	Assignments						
10	Machine Learning	3	2	1	60	20	10	10	--	50	150	4	1	5
11	Image Processingand Pattern Recognition	3	2	--	60	20	10	10	50	--	150	3	1	4
12	Elective -IV	3	2	--	60	20	10	10	50	--	150	3	1	4
13	Data Mining and Knowledge Discovery	3	2	--	60	20	10	10	50	--	150	3	1	4
14	*Project Stage - II		16	--	--	--	--	--	--	150	150	--	8	8
	TOTAL	12	24	1	240	80	40	40	150	200	750	13	12	25

Elective - IV: a) Mobile Computing b) Design Patterns c) Network Management System d) Parallel and Distributed Computing

Total Credits

Semester - VII = 25

Semester –VIII = 25

Grand Total = 50

Distributed Systems

Teaching Scheme	Examination Scheme	Credit Scheme
Theory:3 Hours / Week	End Semester Examination: 60 Marks	Theory: 03
	Internal Assessment: 40 Marks	

Course Pre-requisites:

Data Structures, Operating System, Computer Networking.

Course Objectives:

1. Understand foundation of distributed system
2. Introduce the idea of peer to peer services and file system
3. Understand in detail the system level and support required for distributed system
4. Understand the issues involved in studying process and resource management

Course Outcomes: Upon completion of the course student will able to

1. Discuss trends in distributed Systems.
2. Apply network virtualization.
3. Apply remote method interaction and objects.
   Design process and resource management systems
4. Solve issues related to clock Synchronization and the need for global state in distributed systems.
5. Discuss Pervasive Computing Environment.

Unit I **06 Hours**

Introduction:

Need of Distributed system(DS), Examples of Distributed Systems, Pros and Cons of distributed System, models of DS, Trends in Distributed Systems ,focus on resource sharing, challenges, case study world wide web, System model , Inter process communication, The API for internet protocols, External data representation and multicast communication network, Virtualization, Overlay networks.

Unit II **06 Hours**

Communication in Distributed System :

Case study MPUI remote method Invocation and object : Remote invocation and object : Remote invocation introduction, request reply protocols, remote procedure call, remote method invocation, case study java RMI group communication, publish, subscribe system message queues, shared memory approaches. Distributed objects case study Enterprise java beans from objects to components.

Unit III **06 Hours**

Peer to peer services and file system:

peer to peer system introduction, napster and its legacy peer to peer systems middleware routing overlays case studies: pastry , japestry, distributed file system introduction file service architecture Andrew file system file system features file model file accessing models file sharing semantics Naming identifiers, address Name resolution, Name space Implementation, name cache LDAP.

Unit IV **06 Hours**

Synchronization and replication:

Introduction, clocks, events and process states, synchronizing physical clock, logical time and logical clocks, global states, coordination and agreement, Introduction of distributed mutual exclusion, elections transactions and concurrency control, nested transactions, locks, optimistic concurrency control, time stamp ordering, atomic commit protocols, distributed deadlocks, Replication , Case study CODA.

Unit V **06 Hours**

Process and resource management:

Process management, process migration, features, mechanism, Threads, models, issues, implementation resource management, introduction, features of scheduling algorithms, task assignment approach, load balancing approach, load sharing approach.

Unit VI **06 Hours**

Security:

Main threads and techniques for ensuring security (secure channels, firewalls) Fault tolerance and availability (passive /active replication, gossip architectures) application. Pervasive computing environments: active office, home and city, mobility and location- tracking etc.

Assignment:

1. Study of design and implementation issues of Distributed System.
2. Demonstrate the Mutual Exclusion and Deadlock detection of any Distributed System.
3. Study of CORBA RMI and CORBA-SERVICEs.
4. Design and implementation issues of Distributed Shared Memory (DSM).
5. Design and implementation of any scheduling algorithm of any new DS
6. Study of Security aspects in Distributed System.
7. Case Study NFS and AFS.
8. Study of process management in CODA.
9. Case study : applications of pervasive Computing
10. Study of recourse management in Amoeba.

Textbooks:

1. George Coulouris, Jean Dolimore and Jim Kindberg “Distributed Systems Concepts and Design” fifth edition, Pearson education, 2012.
2. Pradeep K Sinha, “Distributed Operating Systems Concept and Design” Prentice Hall of India .
3. Tanenbaum A.S., Van Steen M “.Distributed Systems, Principles and Paradigms.” Pearson education 2007.

Reference Books:

1. Liu M. L. “ Distributed Computing Principles and Applications,” Pearson Education 2004.
2. Nancy A. Lynch, “Distributed Algorithms” Morgan Kaufman publishers USA.
3. Speciner, Kaufman and Perlman, “Network Security” Pearson Education 2009.

Network Security and Cryptography

Teaching Scheme	Examination Scheme	Credit Scheme
Theory:3 Hours / Week	End Semester Examination: 60 Marks	Theory: 03
Practical:2 Hours / Week	Internal Assessment:40 Marks	TW &Practical:01
	Term Work & Practical:50 Marks	

Course Pre-requisites:

Basic knowledge of computer network.

Course Objectives:

1 To understand basics of cryptography, how it has evolved, and some key encryption techniques.

2 To learn security policies such as authentication, integrity and confidentiality.

Course Outcomes: Upon completion of the course student will able to

1 Understand the basics of network security

2 Learn different techniques of cryptography

3 Discuss details of key and certificate management

4 Learn about system security

5 Recite Network and Transport Layer security

6 Apply knowledge of network security and cryptography in real life

Unit I **06 Hours**

Introduction to network security:

Goals of security, Threat scenarios, Types of attacks: Denial of service, Non-repudiation, Principles of security, A Model for Network Security, Security approaches and policies, Authentication, Authorization and Accounting, Physical and logical access control, User authentication, Biometrics devices, Security services and mechanisms, Privacy and data protection, Cyber laws.

Unit II **06 Hours**

Symmetric and Asymmetric key cryptography:

Symmetric and Asymmetric key cryptography:-Plain Text and Cipher Text, Substitution techniques, Transposition Techniques, Encryption and Decryption, Data Encryption standards, Block Cipher Design Principles ,Advanced Encryption Standards, RC4/RC5 algorithm, IDEA, RSA, Blowfish, S-Box theory, EIGamal Cryptosystem. Steganography, COMSEC, TRANSEC.

Unit III **06 Hours**

Key and Certificate Management

Hash Functions, Digital Signatures, Digital Certificates, Knapsack Algorithm, Certificate & DDOS-Distributed Dos based authentication, Smart Cards, PKIX model, PKI components and Applications, LDAP protocol, Creating Digital certificates using Java, DNS certificates, Key agreement protocols: STS protocol, Diffie- Hellman Key Exchange

Unit IV **06 Hours**

System Security:

Secure Socket Layer, Secure Electronic Transaction, 3-D Secure protocol, Secure HTTP, Time stamping protocol, Email Security, SMTP, PEM, PGP, Wireless Application protocol, Authentication applications- X.509, Kerberos. Key Distribution Centre, Single Sign ON approaches, Security in GSM and 3G.

Unit V **06 Hours**

Security in Network and Transport Layer:

I CMP redirect hazard, ARP hazard, secures network infrastructure services: DNS, NTP, SNMP, Secure RSVP, TCP/IP protocol Suite, IP Security-IP security services, Security Associations. IPv4/IPv6 encapsulation header, IKE protocol, Point to Point Tunneling Protocol, Layer 2 Tunneling Protocol, Replay Attacks, Encapsulating Security Payloads, Data Compression Using Zip.

Unit VI **06 Hours**

Network Security Practices:

Types of firewalls, Firewall Configurations and Implementation, Firewall forensic, Firewall services and limitations, Network address Translation (NAT), DMZ networks, Source masking and hidden channels, VPN Architecture, Intruders, Intrusion Detection, Network based and host based Intrusion Prevention System, Intrusion Detection tools, Architecture for Distributed Intrusion Detection, System Integrity Verifiers, Log File Monitors, Honeypots.

Term Work:

1. Introduction to Cryptography based Security Tools.
2. Write a Program in C/Java to implement symmetric encryption.
3. Write a Program in C/Java to implement asymmetric encryption.
4. Introduction to GnuPG encryption system.
5. Implementation of Decryption techniques using secret key in GnuPG.
6. Implementation of various cryptographic algorithms using HashCalc.
7. Study of how Firewall works in computing.
8. Study of how Antivirus works according to offline or online mode.
9. Implement mini project to develop antivirus application.
10. Case study on cyber security

Assignments:

1. Introduction to security and types of attacks.
2. Discuss Security approaches and policies.
3. Study of any one Symmetric key cryptography algorithm.
4. Explain any one Asymmetric key cryptography algorithm.
5. Explain the concept of digital certificates.
6. Explain email security and it's security protocols.
7. Study of Key agreement protocols
8. Discuss system level security
9. Study of various protocols in network security
10. Study of network security practices

Textbooks:

1. Atul Kahate, "Cryptography and Network Security", 2nd Edition, Tata McGrawHill
2. William Stallings, "Cryptography and Network Security", Pearson Education.

Reference Books:

1. Bruce Schneier ,"Applied Cryptography-Protocols, Algorithms, and Source Code in C".
2. William R. Cheswick. Steven M. Bellovin, Aviel D. Rubin, Addison-Wesley. "Firewalls and Internet Security, Repelling the Wily Hacker".
3. J.W. Rittiaghouse and William M.Hancok – Elseviers. "Cyber Security Operations Handbook".
4. Menezes, van Oorschot and Vanstone, "Handbook of Applied Cryptography".

Syllabus for Unit Tests:

Unit Test -1 UNIT – I, UNIT – II, UNIT – III

Unit Test -2 UNIT – IV, UNIT – V, UNIT – VI

Big Data Analytics and Architecture

Teaching Scheme	Examination Scheme	Credit Scheme
Theory:3 Hours / Week	End Semester Examination: 60 Marks	
	Internal Assessment:40 Marks	

Course Pre-requisites:

Knowledge on Database Management System, Distributed Computing, Data Mining, File System, Statistics.

Course Objectives:

This course focuses on the basic concepts of big data, methodologies for analyzing structured and unstructured data using Hadoop & R.

Course Outcomes: Upon completion of the course student will able to

1 To outline the Big Data fundamentals and various data challenges getting introduced.

2 To accurately and objectively examine, and critically investigate Big Data Technologies and Infrastructure concepts, theories.

3 To apply the novel architectures and platforms introduced for Big data, in particular Hadoop and MapReduce.

4 Being able to describe and apply the Data Analytics lifecycle to Big Data projects

5 To analyze methods and algorithms with respect to data and application requirements, and make appropriate design choices when solving real-world problems.

6 To Understand and apply secured policy oriented techniques for solving real-world problems.

Unit I **06 Hours**

Introduction to Big Data:

Types of Data, Characteristics of big data Information, Phases: capture, organize, integrate, analyze, and act, Defining Structured Data, exploring sources of big structured data, Understanding the role of relational databases in big data, Defining Unstructured Data, exploring sources of unstructured data, Understanding the role of a CMS in big data management, Integrating data types into a big data environment.

Unit II **06 Hours**

Statistics & Analytics:

Numerical Data, Summary Statistics, Sampling and Confidence Intervals, From Population to Sampled Data, Different Types of Biases, Probability, Statistical Inference, Conditional Probability and Bayes Rule, Random Variables, Linear Regression, Bayesian Modeling, , Text Analytics, Characteristics of big data analysis, Linear Models, Decision trees, Neural Networks,

Unit III **06 Hours**

Predictive Analysis Process and R:

Data Analytics Lifecycle, Discovery, Data Preparation, Model Planning, Model Building, communicate results, Operationalize, Building a Predictive model.

Introduction to R, R graphical User Interfaces, Data import and Export, Dirty Data, Data Analysis, Statistical Methods for Evaluation, Linear regression with R, Clustering with R hypotheses testing.

Unit IV **06 Hours**

Big Data Storage and Computing Platforms:

Distributed computing for big data, the changing economics of computing, the problem with latency, Demand meets solutions, Physical infrastructure layer, Big Data Stack, Layer 0: Redundant Physical Infrastructure, Layer 1: Security Infrastructure, Layer 2: Operational Databases, Layer 3: Organizing Data Services and Tools, Layer 4: Analytical Data Warehouses, virtualization in big data, Traditional RDBMS, NoSQL, Design Goals for MongoDB

Unit V **06 Hours**

Hadoop:

The Core of Hadoop: MapReduce, Hadoop's Lower Levels: HDFS and MapReduce, Improving Programmability: Pig and Hive, Common building blocks, Key/value pairs, Getting your data into Hadoop, Other Hadoop Components, Hadoop in action.

Unit VI **06 Hours**

Case Studies and Data Privacy:

Defining Privacy and Security, Data and User Anonymization Fraud and Big Data, Risk and Big data, Credit Risk Management, Big Data & Algorithm Trading, Advertising and Big Data, The Privacy Landscape, Rights and Responsibilities, Case Study: Recommendation Engine, Sentiment Analysis and Digital Marketing, Healthcare applications.

Assignment:

1. Quizzes and Surprise tests will be conducted for testing the knowledge of students for particular topic.
2. Case study on Resilient Distributed Datasets.
3. Case study on Big Data Analytics with Hadoop.
4. Case study on Big Data Analytics with Spark.
5. Case study on how data analytics stacks work and the factors influencing their performance.
6. Discuss in brief the Association analysis.
7. Prepare a case study on Deep Analysis.
8. Illustrate how WEKA can be used for big data Analysis
9. Illustrate and setup a single node Hadoop cluster.
10. Illustrate the best practices for setting-up and using Hadoop.

Textbooks:

Vignesh Prajapati, "Big Data Analytics with R and Hadoop," Packt Publishing Ltd, 2013,

References:

1. John Wiley & Sons, "Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data", EMC Education Services, 2015, 1118876059, 9781118876053
2. Raj, Pethuru, "Handbook of Research on Cloud Infrastructures for Big Data Analytics",, IGI Global, 2014, 1466658657, 9781466658653,
3. Wiley CIO, Michael Minelli, Michele Chambers, Ambiga Dhiraj, John Wiley & Sons, " Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses", 2012, 1118239156, 9781118239155

Syllabus for Unit Tests:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

Compiler Design

Teaching Scheme	Examination Scheme	Credit Scheme
Theory:3 Hours / Week	End Semester Examination: 60 Marks	Theory: 03
	Internal Assessment: 40 Marks	

Course Pre-requisites:

1. The students should have learnt Theory of Computation.
2. Basic of the structure of any Programming Language and Grammars.
3. Know the basics of Computer organization and Assembly Language Programming.

Course Objectives:

1. To study the Compiler Design Tools.
2. To understand the Compiler for various Programming Languages.

Course Outcomes: Upon completion of the course student will able to

1. To have knowledge of LEX and YACC Tools.
2. The students will have the knowledge of Lexical Analysis and formation of TOKENS.
3. Understand Parser and its various techniques.
4. Understand Syntax Directed Translation Scheme
5. Understand the use of Hash Tables and Symbol Tables.
6. Learn the techniques of Code Optimization

Unit I The Phases of a Compiler: (6 Hours)

Translator issues, why to write Compiler, Compilation process in brief, Frontend and Backend Model, Compiler Construction tools, Interpreter and the related issues, Cross Compiler, Incremental Compiler, Boot Strapping, Byte code Compilers.

Lexical Analysis: The role of Lexical Analyzer, Input Buffering, Specification of tokens, recognition of tokens, and language for specifying Lexical Analysis alphabet, Token, Lexical error, Block schematic of Lexical Analyzer, "Automatic Construction of Lexical Analyzer-(LEX), LEX Specification and Features. Pattern Matching Algorithms and their Optimization, use of LEX.

Unit II **06 Hours**

Syntax Analysis Introduction:

Role of Parsers, Review of Context Free Grammar for Syntax Analysis. Top down-RD parser, Predictive parsers, LL (k) parsers, Bottom up Parsers - Operator Precedence Parsers, Shift-Reduce: SLR, LR (k), LALR etc. using Ambiguous Grammars. Error Detection and Recovery,

Automatic Construction of Parsers (YACC), YACC specifications. Canonical LR parsers, handling of ambiguous grammars, Error Reporting in LL (1), Operator Precedence and LR Parsing, Efficient Generation of LALR (1) sets, Optimization of LR parsers, Optimization of transformations, Detection, Reporting, Recovery and Repair of errors in the Compilation Process.

Semantic Analysis Need of Semantic Analysis, Type Checking and type Conversion Overloading of Functions and Operators, Polymorphic Functions, Unification Algorithm.

Unit III **06 Hours**

Syntax Directed Translation:

Syntax Directed Translation Schemes, Implementation of Syntax Directed Translators , Intermediate Code, Postfix Notation, Parse Trees and Syntax Trees, Three-Address code, Quadruples, and Triples, Translation of Assignment Statements, Boolean Expressions, Translation with a Top-Down Parser. Runtime Environment.

Unit IV **06 Hours**

Code Optimization:

Basic blocks and folding, optimization within iterative loops, global optimization through flow graph analysis, Code-Improving Transformations, Machine Dependent Optimization Introduction, Classification of optimization, Principle sources Of Optimization, optimization of basic blocks, Loops in flow graphs, Optimizing transformations: compile time evaluation, Common sub-expression elimination, variable propagation, code movement, strength reduction, dead code elimination and loop optimization, Local optimization, Global Optimization: Control and data flow analysis, Computing Global Data flow information: Meet over paths, Data flow equations, Data flow analysis, Iterative Data Flow Analysis: Available expressions, Live Range Identification.

Unit V **08 Hours**

Code Generation Introduction:

Intermediate languages, Translation of Declarations & Assignments statements. Design issues of a Code generator, Target machine, Runtime storage Management, Basic blocks and flow graphs. Issues in code generation, Target machine description, Basic blocks and flow graphs, next-use information, Register allocation and assignment, Dag representation of basic blocks, Peephole optimization, Generating code from a DAG, Dynamic programming, Semantic stacks, Attributed Translation, Analysis of syntax, Directed Translation, Evaluation of expressions, control structures, Procedure calls.

Unit VI **08 Hours**

Case Study of GCC:

Architecture, Command Line Options for assembly, preprocessing and Intermediate tokens. Optimization Levels of GCC. GCC Help

Assignment:

1. Study LEX and YACC Tools.
2. Divide a 'C' Language Program into Tokens and design a Lexical Analyzer Program to display the list of tokens in it. Display Libraries used, functions and variables defined
3. Study Top down Parser.
4. Study Bottom up Parser.
5. Study Different Syntax Directed Translation Scheme.
6. Represent High Level Language Instruction in the form of 3 address Code, Quadruples and Triples.
7. Study the different methods of Code Optimization.
8. Write a program to generate a Code for C Language Statements.
9. Study of GCC
10. Create a language translator from C to PHP.

Text books:

1. K. Cooper, L, Torczon, "Engineering a Compiler", Morgan Kaufinann Publishers, ISBN 81-8147- 369-8.
2. K. Louden, "Compiler Construction: Principles and Practice", Thomson Brookes/Cole (ISE), 2003, ISBN 981 - 243 - 694-4.
3. J. R. Levine, T. Mason, D. Brown, "Lex & Yacc", O'Reilly, 2000, ISBN 81-7366 -061-X.
4. S. Chattopadhyay, "Compiler Design", Prentice-Hall of India, 2005, ISBN 81-203-2725-X.
5. D. M. Dhamdhere, "Compiler Construction—Principles and Practice", (2/e), Macmillan India.
6. Andrew Appel, "Modern Compiler Implementation in C", Cambridge University press.
7. K C. Louden "Compiler Construction—Principles and Practice" India Edition, CENGAGE.
8. Bennett J.P., "Introduction to Compiling Techniques", 2/e (TMH).

Reference Books:

1. A V Aho, R. Sethi, .J D Ullman, "Compilers: Principles, Techniques, and Tools", Pearson Education, ISBN 81 - 7758 - 590 – 8.

Syllabus for Unit Tests:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

Elective –III a):Grid Computing

Teaching Scheme	Examination Scheme	Credit Scheme
Theory:3 Hours / Week	End Semester Examination: 60 Marks	Theory: 03
	Internal Assessment:40 Marks	

Course Pre-requisites:

Knowledge of distributed systems.

Course Objectives:

1. Understand how Grid computing helps in solving large scale scientific problems.
2. Learn how to program the grid.
3. Understand the security issues in the grid.

Course Outcomes: Upon completion of the course student will able to

1. Understand basic concepts grid computing with the help of case studies.
2. Understand basic architecture of grid.

Unit I **06 Hours**

Introduction to the Grid:

History and Evolution of Computing from Mainframe to Grid, Difference between Grid and Cloud Computing, Introduction to High Performance Computing and Distributed Computing, Definition of Grid, Characterization of the Grid, Architecture of the Grid, Types of Grids, Scope and Applications of Grid Computing Applications. Grid Computing Organizations and Their Roles: Developing Grid Standards & Best Practice Guidelines.

Unit II **06 Hours**

Grid Services and Monitoring :

Introduction to Open Grid Services Architecture (OGSA) – Motivation – Functionality Requirements – Practical & Detailed view of OGSA/OGSI – Data intensive grid service models – OGSA services. Grid Monitoring Architecture.

Unit III **06 Hours**

Grid Development Toolkits:

GlobusGT3 Toolkit :Architecture - Programming Model Implementation-High-Level Services

Unit IV **06 Hours**

Grid Scheduling and Resource Management:

Introduction, Scheduling Paradigms, Working principles of Scheduling, A Review of Condor, SGE, PBS and LSF, Grid Scheduling with QoS.

Unit V **06 Hours**

Grid Security:

Introduction, Cryptography, Grid Security Infrastructure and Authorization Models, Possible Vulnerabilities.

Unit VI **06 Hours**

Grid Portals and Grid Applications:

Introduction, First and Second Generation of Grid Portals, GT3 Use Cases, Resource Management Case Studies, Grid Portal Use Cases.

Assignments:

1. Case Studies on Grid Computing.
2. Case Studies on Performance Analysis.
3. Illustrate difference between Grid Computing & Cloud Computing.
4. Explain Grid Monitoring Architecture in detail.
5. Describe Scheduling Paradigms of Grid Computing.
6. Illustrate difference between Condor, SGE, PBS and LSF.
7. Explain Grid Security Infrastructure.
8. Describe Grid Services.
9. Case study of GlobusGT3 Toolkit.
10. Case study of Resource Management.

Textbooks

1. Joshy Joseph, and Craif Fellenstein, “Grid Computing”, IBM Press, Pearson education, 2011.
2. Maozhen Li, Mark Baker, "The Grid: Core Technologies", ISBN: 978-0-470-09417-4.
3. Paul E. Ceruzzi, "A History of Modern Computing" Second Edition

Reference Books

1. Fran Berman, Geoffrey Fox, Anthony J. G. Hey “Grid Computing: Making the Global Infrastructure a Reality”, John Wiley & Sons.

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

Elective –III b):VLSI and Embedded System

Teaching Scheme	Examination Scheme	Credit Scheme
Theory:3 Hours / Week	End Semester Examination: 60 Marks	Theory: 03
	Internal Assessment:40 Marks	

Course Pre-requisites:

Digital Logic, Analog Circuits, Operating system, System Software, Computer Architecture and Organization, Basic Microprocessors, Theory of Computation, Discrete Mathematics.

Course Objectives:

To make students aware of basic embedded system architecture ,tools for developing embedded systems and to develop ARM programming model

Course Outcomes: Upon completion of the course student will able to

1 Understand the basic architecture of Embedded System.

2 Describe the various embedded system development tools

3 Develop a ARM programming model

4 Explain basic MOS transistor action

5 Analyze Combinational and Sequential Logic

6 Design Mealy and Moore state models using Verilog.

Unit I **06 Hours**

Introduction to Embedded Systems:

Architecture of Embedded System, Design Methodology, Design Metrics, General Purpose Processor, System On chip. Embedded system design and development, Life-Cycle Models, Development tools. System design specifications Functional design, Architectural design, Prototyping.

Unit II **06 Hours**

ARM Architecture:

ARM Design Philosophy, Registers, PSR, Pipeline, Interrupts and Vector Table, Architecture Revision, ARM Processor Families ARM architectural support for operating System, Memory subsystem architecture, Designing a cache system, Memory allocation, Communication protocols.

Unit III **06 Hours**

ARM Programming Model:

Instruction Set: Data Processing Instructions, Branch, Load, Store Instructions, PSR Instructions, Conditional Instructions. Register Usage, Other Branch Instructions, Data Processing Instructions, Single-Register and Multi Register Load-Store Instructions, Stack, Software Interrupt Instructions ,Instruction Scheduling, Register Allocation, Conditional Execution and Loops.

Unit IV **06 Hours**

Introduction to MOS Technology:

Basic MOS Transistor action: Enhancement and Depletion Modes. Basic electrical properties of MOS, Threshold voltage and Body Effect. Design of MOS inverters with different loads, Basic Logic Gates with CMOS: INVERTER, NAND, NOR, AOI and OAI gates. Transmission gate logic circuits, Bi-CMOS inverter.

Unit V **06 Hours**

Combinational Logic:

Manchester, Carry select and Carry Skip adders, Crossbar and barrel shifters, Multiplexer.

Sequential Logic: Design of Dynamic Register Element, Dynamic RAM Cell, Static RAM Cell.D flip flop using Transmission gates. NOR and NAND based ROM Memory Design.

Unit VI **06 Hours**

Introduction to HDLs:

Basic Concepts of Verilog, Data Types, System Tasks and Compiler Directives. Behavioural Modelling: Structured Procedures, Procedural Assignments, Timing control, Conditional statements, Sequential and Parallel Blocks, Generate Blocks. Switch level Modelling. Tasks, Functions, Procedural Continuous Assignments, Design of Mealy and Moore state models using Verilog.

Assignment:

1. Define Embedded system and explain it's architecture
2. Study various tools used in development of embedded systems
3. Analyze various memory allocation algorithms
4. Explain in detail ARM architecture
5. Study instruction set required for developing ARM programming model
6. Explain use of basic logic gates in designing a transmission circuit
7. Distinguish between combinational and Sequential logic
8. Explain in detail various types of of flip-flops
9. Describe in detail concept of Verilog
10. Case study Embedded system

Textbooks:

1. Samir Palnitkar, "Verilog HDL: A Guide to Digital Design and Synthesis", Pearson Education, 2008.
2. Michael D. Ciletti, "Advanced Digital Design with Verilog HDL",PHI, 2005.
3. Kamran Eshraghian, Douglas A. Pucknell, and Sholeh Eshraghian, "Essentials of VLSI circuits and Systems", PHI, 2011.
4. John P. Uyemura, "Introduction to VLSI Circuits and Systems", Modern VLSI Design – Wayne Wolf, 3rd Ed., 1997, Pearson Education.
5. Steve Furber, "ARM System-on-Chip Architecture", Second Edition, Pearson Education Publication.
6. James K. Peckol, "Embedded Systems: A Contemporary Design Tool", WILEY Student Edition Publication.
7. Tammy Noergaard, "Embedded Systems Architecture", Elsevier Publication.

References:

1. "Introduction to VLSI Systems: A Logic, Circuit and System Perspective",Ming-BO Lin, CRC Press, 2011.
2. "Principals of CMOS VLSI Design",N.H.E Weste, K. Eshraghian,, 2nd Ed., Addison Wesley.

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

Elective –III c):Artificial Intelligence and Robotics

Teaching Scheme	Examination Scheme	Credit Scheme
Theory:3 Hours/Week	End Semester Examination: 60 Marks	Theory: 03
	Internal Assessment:40 Marks	

Course Pre-requisites:

Data Structures, Algorithms, Discrete Mathematics,

Course Objectives:

To make student aware of basics of Artificial Intelligence (AI), Knowledge representation methods, learning concept and basics of robotics

Course Outcomes: Upon completion of the course student will able to

1 Understand the basic search algorithms
2 Describe the various knowledge representation strategies
3 Understand plan generation systems
4 Describe various learning methods
5 Understand basic robotics concepts
6 Understand kinematics of robot

Unit I **06 Hours**

Introduction to AI and Production systems:

Definition, Problem formulation Control strategies ,Search strategies, Problem Characteristics, Production systems, Problem Solving methods,- Problem Graphs Indexing, Heuristic functions, Hill Climbing, Best First Search, Minimax Search for two player games Constraints Satisfaction,- Related algorithms, Performance measure and analysis of search algorithms

Unit II **06 Hours**

Knowledge Representation:

Knowledge Representation using predicate logic, Predicate calculus, Resolution, Knowledge representation using predicate calculus Knowledge representation using other logic, Structured representation of knowledge. Production based systems, Frame based systems, Scripts, Conceptual dependency Forward and backward chaining, Rule value approach, Fuzzy reasoning., Bayesian theory

Unit III **06 Hours**

Planning:

Basic plan generation systems, Components of planning system Advanced, Strips plan generation systems, K-strips strategic explanation Planning with state-space search – partial-order planning – planning graphs – planning and acting in the real world

Unit IV **06 Hours**

Learning:

Learning concept, Supervised and unsupervised learning, Learning from observation - Inductive learning – Decision trees – Explanation based learning – Statistical Learning methods - Reinforcement Learning

Unit V **06 Hours**

Robotics an Application of AI :

Brief history, types, classification and usage, Science and Technology of robots, Elements of robots -- joints, links, actuators, and sensors Position and orientation of a rigid body, Homogeneous transformations, Representation of joints, link representation using D-H parameters,

Unit VI **06 Hours**

Kinematics of Robots:

Kinematics of serial robots, Mass and inertia of links, Lagrangian formulation for equations of motion for serial and parallel manipulators, Generation of symbolic equations of motion using a computer, Simulation Kinematics of parallel robots Dynamics of serial and parallel robots , Modeling and analysis of wheeled mobile robots

Assignment:

1. Analyze various search algorithms.
2. Illustrate different knowledge representation strategies with example.
3. Explain in detail Bayesian theory.
4. Describe in detail components of planning system.
5. Describe in detail planning and acting in the real world.
6. Case study learning.

7. Explain in detail the basics of robotics.
8. Describe in detail link representation using DH parameters.
9. Describe in detail kinematics of robotic.
10. Case study robotics.

Textbooks:

1. Stuart Russel and Peter Nurving, "AI-A Modern Approach", 2nd Edition, Pearson Education.
2. Peter Jakson,' Introduction to expert systems",3 rd Edition ,Pearson Education,2007
3. Ashitava Ghoshal "Robotics:Fundamental Concepts and Analysis", Oxford University Press,Second reprint, May 2008.
4. Deepak Khemani , "Artifitial Intelligence", Tata Mc Graw Hill Education ,2013

Reference Books:

1. David Poole, Alan Mackworth, Randy Goebel, "Computational Intelligence:a logical approach", Oxford University Press, 2004.
2. G. Luger, "Artificial Intelligence: Structures and Strategies for complex problem solving", Fourth Edition, Pearson Education.
3. J. Nilsson, "Artificial Intelligence: A new Synthesis", Elsevier Publishers.

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

Elective – III d): Ad-Hoc and Sensor Networks

Teaching Scheme	Examination Scheme	Credit Scheme
Theory:3 Hours / Week	End Semester Examination: 60 Marks	Theory: 03
	Internal Assessment:40 Marks	

Course Pre-requisites:

1. Knowledge of Internet and its working, types of networks.

Course Objectives:

1. Create awareness among the students about the basic fundamentals of Ad-Hoc and Sensor networks.
2. Developing the students with the knowledge of updated networking technology.

Course Outcomes: Upon completion of the course student will able to

1. Understand the basics of ad hoc and sensor networks.
2. Understand the communication mechanism of Ad-Hoc & sensor networks.
3. Learning different routing protocols in Ad-Hoc & sensor networks.
4. Discuss the importance of security in Ad-Hoc & sensor networks.
5. Apply the knowledge about quality of service to Ad-Hoc networks.
6. Learning the structure of Ad-Hoc & Sensor networks by considering real life applications.

Unit I **06 Hours**

Introduction:

Basics of Ad Hoc and Sensor networks, Need of Ad-hoc networks, Types of Ad-hoc networks, Ad-hoc network architectures, Need of sensor networks, Types of sensor networks, Combined approach: Architectures of Ad-hoc & Sensor networks, Working of Ad-hoc & Sensor networks.

Unit II **06 Hours**

Communication in Ad-hoc & Sensor Networks:

Wireless communication technology, information transport through electromagnetic spectrum, role of radio waves in wireless networks, wireless channels, wired channels, internet working, Ad-hoc and Sensor network connections, requirement for Ad-Hoc and sensor network connectivity, wireless LAN configuration, multichannel MAC, IEEE 802.11 standards.

Unit III **06 Hours**

Ad-Hoc & Sensor Network Protocols:

TCP/IP in Ad-hoc networks, MAC protocols: Ad-hoc networks, sensor networks, Routing Protocols: Ad-hoc networks, multicast routing protocols: an architecture reference model for multicast routing protocols, classifications of multicast routing protocols, comparisons of multicast routing protocols.

Unit IV **08 Hours**

Security in Ad-hoc and Sensor Networks:

Need of security in Ad-hoc and sensor network , role of transport layer in ad-hoc and sensor networks, architecture of security layer in Ad- hoc networks, transport layer security protocols, TCP over Ad-hoc wireless networks, security in sensor networks, network security requirements, issues and challenges in security provisioning, network security attacks , secure routing in Ad-hoc and wireless sensor networks, issues in designing a transport layer protocol for Ad-hoc and sensor networks, real life example to understand need of security in Ad-hoc and sensor networks.

Unit V **06 Hours**

QoS and Energy Management :

Classifications of QoS Solutions: MAC Layer, Network Layer, Security, QoS Frameworks for Ad hoc Wireless Networks, energy management in Ad-hoc wireless networks, different schemes, types of energy resources, transmission power management schemes, system power management schemes, energy efficiency and utilization schemes in Ad-hoc and Sensor networks, issues and challenges in providing QoS in Ad-hoc and sensor Networks.

Unit VI **06 Hours**

Applications of Ad-hoc and Sensor Networks:

Mobile Ad-Hoc Network (MANET), Wireless Sensor Networks (WSNs), role and applications of Ad-hoc and Sensor networks in Internet of Things, Ad-hoc and sensor network approach: Zigbee, Raspberry pi, Arduino, Wi-Fi modems, Real life examples of Ad-hoc & Sensor networks, Case studies of Ad-hoc and sensor networks.

Assignments:

1. Explain working of Ad-Hoc networks with considering real life example.
2. Explain working of Wireless Sensor networks with real life example.
3. Explain working of MAC protocol with real life example.
4. Explain different IEEE 802.11 standards.
5. Problems based on Ad-hoc routing/ multicast routing protocols.
6. Program code in C/Java/ other to implement any functionality of Ad-hoc and sensor networks.
7. Explain role of transport layer in Ad-hoc networks.
8. Research proposal of mini project to establish own Ad-hoc & Sensor networks.
9. How the QoS can be maintained while establishing Ad-hoc & Sensor networks.
10. Case study of Ad-hoc & Sensor networks.

Text Books:

1. Carlos De Morais Cordeiro, Dharma Prakash Agrawal "Ad Hoc & Sensor Networks: Theory and Applications", World Scientific Publishing Company, 2006.
2. Adrian McEwen, Hakim Cassimally "Designing the Internet of Things", WILEY, 2013.

Reference Books

1. C. Siva Ram Murthy, and B. S. Manoj, "Ad Hoc Wireless Networks: Architectures and Protocols ", Prentice Hall Professional Technical Reference, 2008.
2. Carlos De Morais Cordeiro, Dharma Prakash Agrawal, "Ad Hoc and Sensor Networks: Theory and Applications", ISBN: 978-81-7596-792-2 Cambridge University Press India Pvt. Ltd.

3. C. K. Toh, “Ad Hoc Mobile Wireless Networks Protocols and Systems”, Prentice Hall
4. Charles E. Perkins, “Ad Hoc Networking”, Addison Wesley, 2000
5. Erdal Cayirci, Chunming Rong, “Security in Wireless Ad Hoc and Sensor Networks”, WILEY, 2009.

Syllabus for Unit Tests:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

Programming Lab- V

Teaching Scheme	Examination Scheme	Credit Scheme
Practical:2 Hours / Week	Term Work & Practical: 50 Marks	TW & Practical: 01

Course Pre-requisites:

If students have any programming experience at all, he should be able to pick up the general R syntax quite easily.

Course Objectives:

To develop ability to use the computational languages necessary for engineering practice

To provide hands- on experience of the recent platform, technologies and design methodologies used in developing applications.

Course Outcomes: Upon completion of the course student will able to

1. Recite R language fundamentals and basic syntax
2. Demonstrate how R is used to perform data analysis
3. Recite major R data structures
4. Illustrate visualizations using R
5. Implement various functions using R studio.
6. Design different Statistical models.

Unit I **06 Hours**

Understanding Big Data and R basics:

Evolution of R, Features of R, Local Environment Setup, R Command Prompt, R Script File, Comments, R –Data Types, R –Variables, Types of Operators, the alternatives to R.

Unit II **06 Hours**

Data structures in R:

Vectors, Vectors and assignment, Vector arithmetic, Generating regular sequences, Character vectors, Index vectors, Lists: Constructing and modifying lists, Matrices ,Arrays, Factors: Ordered and unordered factors, Data Frames: Making data frames, Working with data frames

Unit III **06 Hours**

R programming fundamentals:

Conditions and loops, R Programming, R -If Statement, R –If...Else Statement, The if...else if...else Statement, R –Switch Statement, R -Repeat Loop, R-While Loop, R –For Loop, Loop Control Statements, Functions in R, Built-in Function, User-defined Function, Calling a Function, Objects and Classes, Debugging.

Unit IV **08 Hours**

Working with data in R:

Getting and Setting the Working Directory, Reading data from files (CSV, EXCEL), The read. table() function, The scan() function, Accessing built- in datasets, , Reading text files, Reading XML File, XML to Data Frame,R-JSON file, JSON to a Data Frame Writing and saving data objects to file in R.

Unit V **08 Hours**

Strings and Dates in R:

String operations in R:String Manipulation, Concatenating Strings -paste() function, Extracting parts of a string, Regular Expressions, Dates in R, R – Pie Charts, 3D Pie Chart , R-bar chart, Group Bar Chart and Stacked Bar Chart, use of functions ggplot & ggplot2

Unit VI **08 Hours**

Statistical models in R:

Defining statistical models; formulae, Linear models, Generic functions for extracting model information, Analysis of variance and model comparison, ANOVA tables, Updating fitted models, Generalized linear models, Graphical procedures, OS facilities, Research and industry Applications of R .

Term Work:

1. Introduction to Big Data Analytics.
2. Assignment based on Looping.
3. String Handling.
4. Storing and Retrieving Data Using Vectors and Data frames

5. Class and object using R.
6. Create Relationship Model & Get The Coefficients
7. Case studies: Perform regression analysis on existing datasets.
8. Plot The Pie Chart With Title And Rainbow Colour Pallet.
9. Reading And Writing Excel CSV Text Files
10. Subject Teacher should take one Mini Project in the group of 2 students based on above syllabus.

Textbooks:

1. The R Book, by Michael J Crawley 2nd Edition, wiley
2. Hands-On Programming with R: Write Your Own Functions and Simulations by Garrett Grolemund
3. Beginning R: the Statistical Programming Language, wiley

Reference Books:

1. "Using R for Numerical Analysis in Science and Engineering", Chapman & Hall/CRC, 2014, Victor A,Bloomfield.

Industrial Training

Teaching Scheme	Examination Scheme	Credit Scheme
	End Semester Examination: 50 Marks	Term Work & Oral: 03

Course Pre-requisites:

Professional Skills, Knowledge of core computer engineering subjects.

Course Objectives:

- To provide exposure for the students on practical engineering fields
- To have better understanding of engineering practice in general and a sense of frequent possible problems.
- To develop problem Identification abilities in real world
- To experience use of technology /tools for software development.
- To Identify their skills, values, beliefs, interests and personal abilities to develop the skills.
- To prepare and present a report.

Course Outcomes: Upon completion of the course student will able to

I. Propose a solution to solve real world problems with the help of technology.

II. Apply software engineering principles.

III. Evaluate and compare the various methodologies to solve a real world problem.

IV. Report hands on experience of using modern software development tools.

V. Assess their skills, values, beliefs, interests and personal abilities and act in congruence with them.

VI. Identify social and ethical responsibilities and develop skills to compete for lifelong learning.

As a part of the B.Tech Computer Engineering curriculum, Industrial Training is a Practical course, which the students B.Tech Computer Engineering should undergo in reputed Private / Public Sector / Government organization / companies as industrial training of 45 days weeks to be undergone by the student in the summer vacation after the semester VI. Examination. And Oral examination will be conducted at the end of the semester VII.

The Industrial Training Report:

An Industrial Training report should be prepared by each student.. The report is expected to demonstrate development of practical and professional skills in Engineering through technical experience and application of theoretical knowledge. Development of skills in dealing with people, and communication skills form part of the training experience. Students should seek advice from their employers to ensure that no confidential material is included into the report. The student should be able to present the report to prospective employers,

The following should be observed:

i. Length of training

ii. Preliminary information

iii. iii. Technical report/diary References should be made in the text to books, technical papers, standards etc., used during the training period and should be listed.

iv. Finally, a conclusion should include comprehensive comments on the type and value of experience gained, and how this relates to your professional career.

v. A copy of the report should be submitted to his/her employer, another copy to the Department (through the respective Adviser). Students should also retain a personal copy of the report.

Seminar

Teaching Scheme	Examination Scheme	Credit Scheme
Practical:2 Hours / Week	End Semester Examination: 50 Marks	Term Work & Oral: 01

Course Pre-requisites:
Basics of Software engineering, Knowledge of core computer engineering subjects.

Course Objectives:

- To develop problem Identification abilities in real world.
- To study the use of technology for societal benefits.
- To acquire the knowledge of intellectual topics in computer engineering.
- To prepare and present a document .

Course Outcomes: Upon completion of the course student will able to

I. Describe the broader trends of technological growth in the computer, engineering fields.

II. Interpret that technology has profound impact as a trigger for global change.

III. Develop skills to explore intellectual topics in computer engineering.

IV. Develop better skills to succeed in the career.

V. Understand the responsibility and the of ethics as an engineer

The student has to prepare for the seminar presentation and present it before the group of students It is recommended that a student should meet the guide regularly during the course of the seminar.

The following are the guidelines for the seminar:

- He /She can select a paper from his/her area of interest.
- Recent research papers from any reputed journals like Springer/ACM/IEEE can be selected.
- After selecting the paper, the student has to get approval from the concerned faculty In charge /Seminar guide.
- Students are required to acquire a thorough knowledge on the subject by referring back papers and reference books.
- The student has to prepare a MS power point Preparation of slides and present it in front of group of students from the same class in presence of seminar guide followed by question answer session
- He /She have to write a comprehensive report about the seminar at the end of the semester.

The term work can be accessed based on selection of topic, decorum, communication skill, preparation of presentations/slide and seminar report.

Project Stage-I

Teaching Scheme	Examination Scheme	Credit SchemePractical:
8 Hours / Week	End Semester Examination: 50 Marks	TermWork&Oral: 03

Course Pre-requisites:

Basics of Software engineering, Software testing and knowledge of core computer engineering subjects.

Course Objectives:

- To develop problem solving abilities using mathematics.
- To apply algorithmic strategies while solving problems.
- To develop time and space efficient algorithms.
- To develop software engineering documents and testing plans.
- To use algorithmic solutions using distributed, Embedded, concurrent and parallel environments.

Course Outcomes: Upon completion of the course student will able to

I. Review and understand how previous experiences had an impact on affective states and intellectual performance.

II. Identify and define the problem.

III. Decide critically to solve the problem.

IV. Demonstrate the ability to synthesize complex information from a variety of sources in decision-making.

V. Predict and develop a group process and desired outcomes.

VI. Plan and perform collaboratively towards a common purpose.

1. The project will be undertaken preferably by a group of at least3- 4 students who will jointly work and implement the project over the academic year. The work will involve the design of a system or subsystem in the area of Computer Engineering.
2. If the project is chosen a hardware project it will involve the designing a system or subsystem or upgrading an existing system. The design must be implemented into a working model with necessary software interfacing and a user manual.
3. If the project is chosen in the pure Software Application it must involve the detail Software Design Specifications, Data Structure Layout, File Design,

Testing with complete documentation and user interface, with life cycle testing and as an executable package.

4. The group will select a project with the approval of the guide (Staff- members assigned) and submit the name of the project with a synopsis of 2 or 3 pages in the month of August in the academic year. A preliminary study report by the group must be submitted and certified at the end of seventh Semester.
5. It is expected that at least one research paper is published by each group with guide.

The project report stage-I will contain the details

Problem definition and requirement specification, acceptance test procedure (ATP).

a) System definition, requirement analysis.
b) System design with UML.
c) Documentation and references.

Documentation will use UML approach with Presentation, Category, Use Case, Class Diagrams, etc.

Machine Learning

Teaching Scheme	Examination Scheme	Credit Scheme
Theory:3 Hours / Week	End Semester Examination: 60	Theory: 04
Practical:2 Hours / Week	Internal Assessment: 40	TW &Oral:01
Tutorial:1 Hour / Week	Term Work & Oral: 50	

Course Pre-requisites:

Artificial intelligence, Discrete Mathematics, Database Management System, Engineering Mathematics, Programming Languages.

Course Objectives:

To provide a strong formal foundation of Machine Learning concepts and techniques

Course Outcomes: Upon completion of the course student will able to

1. Explain significance of Machine Learning
2. Distinguish between paradigms of Machine Learning.
3. Illustrate use of algorithms in Linear and Non-Linear machine learning.
4. Build Supervised Linear Regression Model.
5. Analyze performance of Supervised and Unsupervised Learning
6. Analyze various ensemble methods and machine learning tools..

Unit I **06 Hours**

Introduction:

Introduction to statistics, Introduction to Learning Systems, Structure of Learning System, Testing vs Training, learning vs Designing, Goal and Applications of Machine Learning, Examples of Machine Learning Problems, Need of Learning, Machine Intelligence

Unit II **06 Hours**

Machine Learning Techniques:

Introduction to Machine Learning Techniques: Supervised Learning(SL) Vs Semi Supervised Learning(SSL) vs Unsupervised Learning(USL), Examples of SL, SS, and US Learning, how to choose Machine Learning Technique, Machine Learning Models, and Types, Examples: Linear based Models, Logic Based and Algebraic Models, Probabilistic Models

Unit III **06 Hours**

Classification:

What is Classification?, Types: Naive Bayes Classifier, Decision Trees, Support Vector Machines, Rule based Classification, Backpropagation, Associative Classification, Classifier Accuracy Measures, Precision and Recall Measures.

Unit IV **06 Hours**

Regression:

What is Regression? Types: Linear Regression, Logistic Regression, Classification vs Regression, Issues Regarding Classification, and Regression, Assessing performance of Regression, Predictor error measures, Applications of Supervised Learning.

Unit V **06 Hours**

Unsupervised Learning:

Introduction to Clustering, Types: K Means Clustering Algorithm, Mixture Models, Hierarchical Clustering, Anomaly Detection, Neural Networks, Self-Organizing Map(SOM), Applications of Unsupervised Learning.

Unit VI **06 Hours**

Trends in Machine Learning:

Ensemble methods for increasing accuracy: Bagging and Boosting, multitask learning, online learning and Sequence Prediction, Data Streams and Active Learning, Introduction to Deep Learning and Reinforcement Learning, Case Study: Latest Machine Learning Tools.

Term Work:

1. Distinguish between Supervised and Unsupervised Machine Learning.
2. Implement Linear and Nonlinear Learning models.
3. Implement Distance based Learning techniques.
4. Write study assignment to build tree based models.
5. Write study assignment to build rule based models.
6. WEKA: Tool for Machine Learning.
7. SCIKIT-LEARN: Tool for Machine Learning.
8. SHOGUN: Tool for Machine Learning.
9. ACCORD: Tool for Machine Learning.
10. Study assignment on Reinforcement Learning technique.

Assignments:

1. Distinguish between Learning and Designing with example.
2. Explain in detail Need of Machine Learning.
3. Explain in detail How to choose Machine Learning Algorithm.
4. Differtiate between different Machine Learning Models.
5. Explain in detail how Classification by Decision Tree Induction.
6. Explain in detail Support Vector Machine for Supervised Learning.
7. Distinguish between Classification and Regression with suitable example.
8. Demonstrate the use of Linear Regression for Supervised Machine Learning.
9. Implement K-Means Clustering Algorithm for Unsupervised Machine Learning.
10. Explain in detail Anomaly Detection.
11. What are the methods used to increase accuracy of machine learning algorithms? Explain.
12. Define Reinforcement Learning with example.

Text books

1. Jiawei Han, Jian Pei, Micheline Kamber, " Data mining concepts and techniques", 3rd Edition.
2. K.P. Soman, R. Loganathan, V. Ajay, "Machine Learning with SVM and Other Kernel Methods"
3. Witten I.H. Author, "Data Mining Practical Machine Learning Tools and Techniques" 2nd Edition.
4. Peter Flach, Machine Learning: The Art and Science of Algorithms that Make Sense of Data", Cambridge University Press, Edition 2012.
5. Hastie, Tibshirani, Friedman, "Introduction to Statistical Machine Learning with Applications in R", Springer, 2nd Edition-2012.

References:

1. T. M. Mitchell, "Machine Learning", McGraw Hill.
2. C. M. Bishop ," Pattern Recognition and Machine Learning", Springer 1st Edition-2013.
3. Ethem Alpaydin, "Introduction to Machine Learning"

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT –II , UNIT –III.

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

Image Processing and Pattern Recognition

Teaching Scheme	Examination Scheme	Credit Scheme
Theory:3 Hours / Week	End Semester Examination: 60	Theory: 03
Practical: 2 Hours / Week	Internal Assessment: 40	TW& Practical: 01
	Term Work & Practical: 50	

Course Pre-requisites:

Set theory, Linear algebra and statistics, Computer Graphics and visualization, Signals and system, Digital signal processing.

Course Objectives:

1 Students should be able to understand digital image processing and advanced concepts.

2 Students should be able to properly implement algorithms using modern computing tools such as MATLAB, and to interpret and present the results.

3 To study fundamentals of colour Image Processing

Course Outcomes: Upon completion of the course student will able to

1 To explain the digital image processing and digital image formation.

2 To illustrate different mathematical preliminaries to deal with digital image processing

3 To explain the concept of Image restoration and image segmentation.

4 To apply the concept of pattern recognition and its different phases

5 To apply knowledge/ skills for solving real world problems.

Unit I **06 Hours**

Digital Image Fundamentals:

Introduction, Fundamental steps and Components of Digital Image Processing, Image Sampling and Quantization: Basic concepts in Sampling and Quantization, Representing Digital images, Spatial and intensity resolution, Relationship between Pixels, Histogram Processing: Definition, Histogram Equalization,

Unit II **06 Hours**

Image Enhancement:

Fundamentals of Spatial Filtering- The Mechanics of Spatial Filtering, Generating Spatial, Filter Masks, Noise Model, Smoothing Spatial Filters: Linear filters – Mean filters Non-linear (Order Statistic filters): Median, Mode, Max, Min filters, Image Enhancement by Frequency Domain Methods: Basic steps for Filtering in Frequency Domain, Frequency Domain low pass (Smoothing),High pass (Sharpening)

Unit III **06 Hours**

Image Compression and Segmentation:

Fundamentals: Coding Redundancy, Spatial and temporal

(Interpixel) Redundancy, Some Basic Compression Methods: Lossless Compression methods-Huffman coding, LZW coding, Fundamentals : Point , Line and Edge Detection, Line Detection, Edge Models, Basic Edge detection, Canny edge detector Thresholding : Foundation, Basic Global Thresholding, Optimal global thresholding, Multiple Thresholds.Region based segmentation: region growing, region splitting and merging.

Unit IV **06 Hours**

Morphological Image Processing and Color Image Processing: Morphological Image Processing, Preliminaries, Erosion and Dilation, Opening and Closing, The Hit-or-Miss Transformation, Some Basic Morphological Algorithms: Boundary Extraction, Hole (Region) Filling, thinning, Color Image Processing: Color Fundamentals and Color Models Basics of Full-Color Image Processing, Color Transformations.

Unit V **06 Hours**

Basics of Pattern Recognition:

Introduction and examples, Clusteringvs. Classification; Supervised vs. unsupervised, Decision Boundaries, Decision region / Metric spaces/ distances, Object detection.

Unit VI **06 Hours**

Clustering and Classification:

Bayes decision rule, Error probability, Linear Discriminates Function (equal covariance matrices)and non- Linear Decision Boundaries (unequal covariance matrices). Clustering: Basics of Clustering; similarity / dissimilarity measures; clustering criteria. Minimum within cluster distance criterion. K-means algorithm, K- medoids, DBSCAN-Density-based Spatial clustering of applicationwith Noise.

Term Work:

1. Display of Grayscale Images.
2. Write a MATLAB code that reads a gray scale image and generates the flipped image of original image.
3. To enhance contrast using Histogram Equalization
4. Write a program for image enhancement.
5. Write a program for image compression
6. Write a program for Edge detection
7. Write a program for image segmentation
8. Write a program for image morphology
9. Illustrate and discuss use of various method of pattern recognition.
10. Write a program for face detection in MATLAB.

Assignments:

1. Write and explain concepts of histogram processing.
2. Write and explain edge detection and Line detection.
3. Write in details about Filtering in Frequency Domain.
4. Write any two algorithms for segmentation.
5. Write in detail about colour image processing.
6. Write and explain concepts of object detection.
7. Write and explain K-means algorithm.
8. Write and explain DBSCAN.
9. Case study on radiographic images to reduce noise in image.
10. Case study on pattern recognition.

Text Books

1. Rafael C Gonzalez, Richard E Woods, “Digital Image Processing”, Pearson Education ,2008.
2. S.Jayaraman,S Esakkirajan,T Veerakumar “Digital Image Processing”, McGrawHill Publication,2009.
3. R.O.Duda, P.E.Hart and D.G.Stork, “Pattern Classification”, John Wiley.

Reference books

2. S.Theodoridis and K.Koutroumbas," Pattern Recognition", 4th Ed., Academic Press, 2009.

3. Anil K Jain, "Fundamentals of Digital Image Processing", PHI

4. B Chanda & Dutta Majumdar, "Digital Image Processing and Analysis", PHI

5. Rafael C Gonzalez, Richard E Woods, Eddins, "Digital Image Processing using MATLAB", Pearson Education

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

Elective-IV a): Mobile Computing

Teaching Scheme	Examination Scheme	Credit Scheme
Theory: 3 Hours / Week	End Semester Examination: 60 Marks	Theory: 03
Practical: 2 Hours / Week	Internal Assessment:40 Marks	TW & Practical: 01
	Term Work & Practical: 50 Marks	

Course Pre-requisites:

Computer Network , Fundamentals of Data Communication.

Course Objectives:

1. Explain the basic of mobile telecommunication system
2. Choose the required functionality at each layer
3. Application identity solution for each functionality at each layer
4. To apply the concept of pattern recognition and its different phases
5. Use similar tool and design Ad-hoc networks.
6. Develop mobile application

Course Outcomes: Upon completion of the course student will able to

1. Explain the basic of mobile telecommunication system
2. Choose the required functionality at each layer
3. Application identity solution for each functionality at each layer
4. Use similar tool and design Ad-hoc networks
5. Develop mobile application

Unit I **06 Hours**

Introduction:

Mobile computing: mobile computing compared with wireless networking, mobile computing applications, characteristics of mobile computing, structure of mobile computing application: MAC protocol, wireless MAC issues, fixed assignment schemes, Random assignment schemes, and reservation based schemes.

Unit II **06 Hours**

Layout Design and Tools:

Transistor structures, Wires and Vias, Scalable Design rules, Layout Design tools. Logic Gates & Layouts:

Static Complementary Gates, Switch Logic, Alternative Gate circuits, Low power gates, Resistive and Inductive interconnect delays.

Unit III **06 Hours**

Mobile Telecommunication System:

Global system for mobile communication (GSM), General packet radio service(GPRS),universal mobile telecommunication system(UMTS) Mobile technology generation, comparison between GMS vs UMTS vs 3G vs 4G vs 5G

Unit IV **06 Hours**

Mobile AD-HOC Network:

Ad –Hoc basic concept, characteristic, Application, Design issues, Routing, Essential of traditional routing protocol, popular routing protocol, vehicular Ad-hoc networks (VANET), MANET VS VANET- security.

Unit V **06 Hours**

Mobile Operating System & Application:

Mobile device operating systems, special constrains & requirements, commercial mobile operating system, software development kit, iOS, iOS architecture, Android, Android architecture, How to develop application via Android, compiling & executing programs in Android, blackberry, window phone.

Unit VI **06 Hours**

Mobile Payment System using Mcommerce:

Mcommerce- structure, pros & cons, mobile payment system, security issues, application issues, GSM mobility management administration.

Term Work:

1. Explain features of mobile IP & mechanism in mobile IP.
2. Explain comparison between GMS vs UMTS vs GPRS.
3. Explain comparison between 3G vs 4G vs 5G.
4. Develop small application in Android.
5. Develop small application in iOS.
6. Simulate the working of Android networking/Communication entities using software tools.

Assignments:

1. Discuss characteristics and applications of mobile computing
2. Explain structure of mobile computing application.
3. Describe TCP/IP architecture in detail.
4. Explain GPRS system with diagram.
5. Prepare a case study on Mobile OS : Android
6. Discuss in detail the implementation of TCP/IP stack in Mobile networks
7. Prepare a caste study on MANET & VANET
8. Prepare a case study on Mobile OS : IOS
9. Prepare a case study on GSM architecture.
10. Prepare a case study on Mobile Payment System.

Text books:

1. Prashant Kumar Patnaik, Rajib mall , "Fundamental of mobile computing", PHI learning Pvt. Ltd.,2012
2. Jochen h schiller, "Mobile Communication", second edition Pearson education 2007

References:

1. Dharma Prakash Agarwal ,Qing and An Zeng, "Introduction to wireless and mobile system", Thomson Asia Pvt. Ltd. 2005
2. Ume Hansmann, Luther mark martin s, Nicklons & Thomas Strober, "principles of mobile computing " ,springer,2003
3. William C.Y.Lee, "Mobile cellular telecommunication Analog and Digital systems", second edition, Tata Mc Graw Hill Edition 2006
4. C. K. Toh, "AdHoc Mobile wireless Networks", first Edition, Pearson Education 2002

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

Elective – IV b): Design Patterns

Teaching Scheme	Examination Scheme		Credit Scheme	
Theory:03 Hours / Week	End Semester Examination: 60 Marks		Theory:	03
Practical: 02 Hours / Week	Internal Assessment:40 Marks		TW & Practical :	01
	Term Work & practical:	50 Marks		

Course Pre-requisites:

Aware about basic java programming concepts.

Course Objectives:

1. To understand the concept of patterns and the catalog
2. To discuss the Presentation tier design patterns and their affect on: sessions, client access

Course Outcomes: Upon completion of the course student will able to

1. To demonstrate a thorough understanding of patterns and their underlying principles
2. To know what design pattern to apply to a specific problem
3. To demonstrate what tradeoffs need to be made when implementing a design pattern
4. To use design patterns for developing software.
5. To understand the variety of implemented bad practices related to the Business
6. To learn how to use design patterns to keep code quality high without overdesign.

Unit I **06 Hours**

Introduction To Design Patterns:

Introduction to design patterns: Design Pattern Definition, Strategy,

Observer, Factory, Singleton, Command, Adapter, Facade, Template Method, Iterator, Composite, State, Proxy Design Patterns in Small Talk MVC, Describing Design Patterns, Organizing the Catalog, Solving of Design Problems using Design Patterns, Selection of a Design Pattern, use of Design Patterns.

Unit II **06 Hours**

Designing A Document Editor:

Design problems, Document structure, Formatting, Embellishing the User Interface, Supporting Multiple Look and Feel standards, Supporting Multiple Window Systems, User Operations, Spelling Checking and Hyphenation.

Unit III **06 Hours**

Design Patterns Catalog:

Creational Patterns, Abstract Factory, Builder, Factory Method, Prototype, Singleton. Discussion of Creational Patterns.

Unit IV **06 Hours**

Structural Patterns:

Adapter, Bridge, Composite, Decorator, Façade, Flyweight, Proxy, Discuss of Structural Patterns

Unit V **06 Hours**

Behavioral Patterns:

Behavioral Patterns- Chain of Responsibility Command, Interpreter, iterator, Mediator, Observer, State, Strategy, Template Method, Visitor, Discussion of Behavioral Patterns, Expectations from Design Patterns.

Unit VI **06 Hours**

Case Studies:

The World Wide Web - a case study in interoperability, Air Traffic Control – a case study in designing for high availability, Celsius Tech – a case study in product line development.

Term Work:

1. Review the engineering design for the part of the product that you must implement. (Engineering Design)
2. Create a new project for this assignment and create a package within the project named scoring.
3. Implement creational pattern.
4. Implement the Score interface. (Design Specifications)

5. Implement the Abstract Score class. (Design Specifications)
6. Implement the Leaf Score class. (Design Specifications)
7. Implement the Composite Score class. (Design Specifications)
8. Solve any two Design Problems using Design Patterns.

Assignments:

1. A journal on PCs would like to investigate some properties of different PCs, e.g. the price, performance, etc. The results need to be sorted in a list. There are several sorting algorithms for sorting all the PCs depending on specific characteristics. These sorting algorithms behave differently with respect to the time and space performance. Therefore, it is required to select and switch to a different sorting algorithm at run-time.
   a) Which design pattern can be applied to fulfill these requirements?
   b) Draw a class diagram that incorporates this pattern.
2. In the future it is expected that class Computer Equipment will be extended with additional operations. The present structure of computer equipment is assumed not to change. It is also not desired to 'pollute' all the existing component classes with new operations.
   a) Which design pattern would be suitable for this problem?
3. Study on Template Method in design pattern
4. Study on Proxy Design Patterns in Small Talk MVC.
5. Solve the Design problems using design patterns.
6. Study on Behavioral pattern.
7. Designing a Document Editor for any system
8. Comparison of design pattern catalogs
9. Study of various structural patterns.
10. Case study on Interpretability.

Text books

1. Gamma, Belm, Johnson ,"Design Patterns: Elements of Reusable Object Oriented Software," PEA.
2. Eric Freeman, "Head First Design Patterns", Oreilly.

Reference books

1. Cooper "Java Design Paterns", , Pearson.
2. Horstmann, "Object Oriented Design and Pattetrns", Wiley.
3. Ali Bahrami, "Object Oriented Systems Development", MCG.

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

Elective-IV c): Network Management System

Teaching Scheme	Examination Scheme	Credit Scheme
Theory:3 Hours / Week	End Semester Examination: 60 Marks	Theory: 03
Practical: 2 Hour / Week	Internal Assessment:40 Marks	TW & Practical: 01
	TW & Practical: 50 Marks	

Course Pre-requisites:

Computer Network, Discrete Mathematics.

Course Objectives:

1. To help students to understand the principles of network management.
2. To provide knowledge about the various network management strategies and network management protocols.

Course Outcomes: Upon completion of the course student will able to

1. To understand network management architectures and protocols.
2. To gain knowledge about basic components of network management.
3. To gain the information about management communication patterns
4. To understand the protocols associated with network management
5. To gain the knowledge about remote monitoring
6. To understand network management metrics

Unit I **06 Hours**

Introduction to Network Management:

Importance of network management, The Players: Different Parties with an Interest in Network Management, Case studies of Networking and Management, Challenges of Information Technology Managers, Current Status and future of Network Management,

The Network Operator's Arsenal:

Device Managers and Craft Terminals, Network Analyzers, Element Managers, Management Platforms , Collectors and Probes

Unit II **06 Hours**

The Basic Ingredients of Network Management:

The Network Devices and its types with Configuration, The Management System, The Management Network, The Management Support Organization: Network Operations Center (NOC).

Unit III **06 Hours**

Management Communication Patterns:

Layers of Management Interactions, Manager Initiated Interactions: Request and Response, Configuration Operations, Agent Initiated Interactions: Events and Event-Based Management.

Unit IV **06 Hours**

Common Management Protocols:

SNMP (Simple Network Management Protocols): Classic and Perennial Favorite , SNMP Operations , SNMP Messages and Message Structure , SNMPv2/ SNMPv2c , SNMPv3, The SNMP Communication Model, Functional model, SNMPv2 Protocol, Major Changes in SNMPv2, SNMPv2 System architecture, SNMPv2 Structure of Management, Information, The SNMPv2 Management Information Base, SNMPv2 Protocol, Compatibility with SNMPv1.

Unit V **06 Hours**

SNMP Management: RMON (Remote Monitoring):

Remote Monitoring Basics, RMON SMI (Structure of Management Information) and Management Information Base (MIB), RMON1, RMON2, ATM Remote Monitoring, A Case Study of Internet Traffic Using RMON. Telecommunications Management Network(TMN):

Fundamentals of TMN, TMN Conceptual Model, TMN Standards, TMN

Architecture, TMN Management Service Architecture, An Integrated View of TMN, Implementation Issues.

Unit VI **06 Hours**

Management Metrics: Assessing Management Impact and Effectiveness:

Network Management Business Impact, Factors that Determine Management Effectiveness, Assessing Network Management Effectiveness.

Web-Based Management: NMS with Web Interface and Web-Based Management, Web Interface to SNMP Management, Embedded Web-Based Management

Term Work:

1. Installation and study of Packet Analyzer tool Wireshark.
2. Installation and study of Honeypots tool.
3. Installation and study of Ethereal tool.
4. Installation and study of Wi-Fi manager.
5. Installation and study of open source Network Management platform (Network Management Information System).
6. Installation and study of Icinga 2 (Open source network monitoring tool).
7. Installation and study of Zabbix.
8. Configure SNMP Protocol on Cisco Packet Tracer.
9. Configure VLAN on Cisco Packet Tracer.
10. Installation and study of Packet Sniffer.

Assignments:

1. Explain the significance of network management system.
2. Describe the components of network management.
3. Justify the role of device Managers and craft terminals in The Network Operator's Arsenal
4. Explain the layers of management interactions.
5. Illustrate SNMP Messages and Message Structure
6. Explain SNMP communication model
7. Explain the improvement in SNMPv2 over the SNMPv1
8. Explain the RMON (Remote Monitoring) with SMI (Structure of Management Information) and Management Information Base (MIB)
9. Describe the TMN (Telecommunications Management Network) architecture.
10. Explain the NMS(network management system) with the context of Web Interface and Web-Based Management

Text Books

1. Mani Subramanian, “Network Management: Principles and Practice”, sixth edition Pearson Education.
2. Alexander Clemm, “Network Management Fundamentals ”
3. D.C. Verma, “Principles of Computer Systems and network Management”
4. Morris,Network management, Pearson Education.
5. Mark Burges, Principles of Network System Administration, Wiley Dreamtech.

Reference books

1. William Stallings, “SNMP, SNMPv2, SNMPv3, and RMON 1 and 2” third edition, Addison-Wesley Professional
2. William Stallings, “Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud”, Addison- Wesley Professional

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

Elective – IV d): Parallel and Distributed Computing

Teaching Scheme	Examination Scheme	Credit Scheme
Theory:3 Hours / Week	End Semester Examination: 60 Marks	Theory: 03
Practical:2 Hours / Week	Internal Assessment:40 Marks	TW & Practical: 01
	Term Work & Practical: 50 marks	

Course Pre-requisites:

C/C++/Java Programming Languages, Data Structures, Linux/Unix Operating System, Distributed Computing, Computer Organization.

Course Objectives:

To focus on the principles of parallel and distributed computing environment and the implementation and performance issues associated with them.

Course Outcomes: After completion of course, students will able to:

1. Ability to recite, explain and model the fundamental concepts and reasoning principles for parallel and distributed systems.
2. Ability to Identify and describe limitations and Challenges of Parallel and Distributed Systems.
3. Ability to adapt and apply the architectural models for various real time applications.
4. Ability to design, analyze algorithms for execution in parallel and distributed settings
5. Ability to report and account for models, limitations, and fundamental concepts in the area of message passing and shared memory concurrency, and apply this understanding to example systems and algorithms.
6. Ability to Outline and assess the significance of high performance computing and its impact in a Computer environment.

Unit I **06 Hours**

Fundamental of Parallel & Distributed Computing:

Introduction to Parallel Processing Paradigms, Modeling and Characterizing Parallel Algorithms, Balanced Trees, Divide and Conquer, Partitioning, Combining, Language Categories and Parallel Programming Languages.

Introduction to Distributed Computing: Computation Model, Client- Server Systems, Peer to Peer Systems, Modularity, Message Passing, Messages on worldwide web.

Unit II **06 Hours**

Promises and Challenges of Parallel and Distributed Systems:

Cost vs. Performance Evaluation, Software and General-Purpose PDC, Turing Machine as the Basis, and Consequences, Complexity Measures for Parallelism, Complexity Measures for Distributed Systems, Processing Technology, Networking Technology, Software Tools and Environments, Neural Networks and Complexity Issues, Tolerating Processor Failures in Synchronous Systems, Tolerating Processor Failures in Asynchronous Systems, Wait-Free Implementations of Shared Objects.

Unit III **06 Hours**

Parallel and Distributed Architectures:

Computational Model, Engineering Model, RISC Architectures, Superscalar and VLIW Processors, SIMD-Processing: Concepts and Systems, MIMD Architectures: Shared and Distributed Memory Designs, Memory Hardware Technology, Memory System Architecture, User-Level Memory Models, and Memory Consistency Models.

Unit IV **06 Hours**

Algorithms & Data Structure for Parallel Programming:

Arrays and Balanced Binary Trees, Linked Lists, Euler Tour vs. Parentheses String, Priority Queues (Heaps), Search Trees/Dictionaries, Impact of Data Distribution, CU/PE Overlap, Parallel Reduction Operations, Parallel Graph Algorithms, Parallel Computational Geometry.

Unit V **06 Hours**

Communication and its Framework:

Message-Passing Model, Distributed Shared Memory Model, Message- Passing System: Desirable Features, Socket-Based Message Passing, p4, Parallel Virtual Machine, Message-Passing Interface(MPI), Separation of Data and Control Functions.

Directory-Based Cache Coherence, Shared Memory Consistency Models, Distributed Memory Architectures, Basic Model: RMI, CORBA, DCOM, Comparison of the Three Paradigms.

Unit VI **06 Hours**

Applications, Tools & Technologies:

HPC, Mixed-Mode Systems, Examples of Existing Mixed-Machine Heterogeneous Computing Systems, Overview of Clustering, Distinct Uses of Clusters, General-Purpose Parallel Computers, Optical Communication: Free-Space Interconnection, Considerations in Choosing the Interconnection Topology, Broadband Integrated Service Data Network (B-ISDN), Parallel and Distributed Computing Environment Over ATM, Parallelizing Compilers and Preprocessors, Performance Monitoring and benchmarking tools, Performance Visualization, Case Studies: Molecular Modeling, Genetic and Protein Sequence Data.

Term Work:

1. To implement a program that plays Conway's Game of Life. Conway's Game of Life is an example of discrete event simulation, where a world of entities live, die, or are born based on their surrounding neighbors. Each time step simulates another round of living or dying.
2. To implement parallel matrix multiply using Pthreads and evaluate the scalability of your implementation as you increase the problem size and the number of threads.
3. Client-server socket programs - To Design a multi-threaded server, using signals, and learning about the HTTP protocol.
4. To Study about CUDA. Implement a forest fire simulator using CUDA.
5. To Understand Locality, Load balancing, and Synchronization Effects using p-Threads.
6. To analyze the tradeoffs among different synchronization algorithms in terms of their latency, fairness, scalability, traffic, and storage requirements.
7. To understand the role of data communication using MPI as the message passing model. Setup the Environment and use the program of Gaussian Elimination.
8. To write a word count program on map-reduce framework.
9. To write a Jacobi Method to solve a system of linear equations using Map-reduce method.

Assignments:

1. To study about Hadoop Architecture.
2. To Study about Spark Architecture.
3. Prepare a case study on CUDA supporting Parallel programming and distributed application.
4. Case study on how to use Java-based TCP communication through a design of chat client and server program.
5. Case study on how to write a parallel-computing application using MPI Java
6. Case study on MPI
7. Note on comparing RMI and mobile agents in terms of programmability and performance.
8. Case study on design and implementation of a very simple distributed file system.
9. Prepare a presentation and case study on A Modern Multi-Core Processor: Forms of Parallelism + Understanding Latency and BW
10. Prepare a presentation and case study on Workload-Driven Performance

Text Books:

1) "Parallel and Distributed Computing: A Survey of Models, Paradigms and Approaches", A Wiley- Interscience publication, Volume 12 of Wiley Series on Parallel and Distributed Computing, Claudia Leopold, 0471358312, 9780471358312.

Reference books

1. "Tools and Environments for Parallel and Distributed Computing, Salim Hariri, Manish Parashar, Volume 34 of Wiley Series on Parallel and Distributed Computing", John Wiley & Sons, 0471474843, 9780471474845
2. "Parallel and Distributed Computing: Theory and Practice. Springer Science & Business Media", 3540580786, 9783540580782.
3. Péter Kacsuk, Dieter Kranzlmüller, Zsolt Németh, Jens Volkert,"Distributed and Parallel Systems: Cluster and Grid Computing, Volume 706 of The Springer International Series in Engineering and Computer Science", Springer Science & Business Media, 2012, 1461511674, 9781461511670
4. Jacek Błażewicz, Klaus Ecker, Brigitte Plateau, Denis Trystram, "Handbook on Parallel and Distributed Processing, International Handbooks on Information Systems", Springer Science & Business Media, 2013, 3662043033, 9783662043035

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

Data Mining and Knowledge Discovery

Teaching Scheme	Examination Scheme	Credit Scheme
Theory:3 Hours / Week	End Semester Examination: 60 Marks	Theory: 03
Practical: 2 Hours / Week	Internal Assessment:40 Marks	TW &OR: 01
	Term Work & Practical:50 Marks	

Course Pre-requisites:

Knowledge of Statistics, Database Management System.

Course Objectives:

1. Identify the scope and necessity of Data Mining & Warehousing for the society.
2. Describe the designing of Data Warehousing so that it can be able to solve the root problems.
3. Understand various tools of Data Mining and their techniques to solve the real time problems.
4. Develop ability to design various algorithms based on data mining tools.

Course Outcomes: Upon completion of the course student will able to

1. Define, describe, and clearly state the objectives of Knowledge Discovery and Data Mining.
2. Describe data mining algorithms
3. Suggest appropriate solutions to data mining problems
4. Analyze data mining algorithms and techniques
5. Identify and or develop software to execute the specified algorithm(s)/data mining technique(s)
6. Identify and distinguish data mining applications from other IT applications

Unit I **06 Hours**

Introduction to Knowledge Discovery and Data Mining:

Data mining fundamentals, Classification of Data mining systems, Integration of Data mining system with Data warehouse, Data Preprocessing,KDD Fundamentals, KDD Process, Data clearing, Data integration and transformation, Data Reduction, Data mining Statistics.

Unit II **06 Hours**

OLAP:

Characteristics of OLAP system, Multidiamensional view and data cube, Data Cube implementations and operations, Difference between OLAP, OLTP and OLAP Server-ROLAP, MOLAP, HOLAP Queries.

Unit III **06 Hours**

Association Rule Mining:

Introduction, The Task and Naive Algorithm,Apriori Algorithm,Improving the efficiency of Apriori algorithm, Direct hashing and pruning(DHP),Dynamic Item set counting(DIC),Mining frequent patterns without candidate generation(FP Growth).

Unit IV **06 Hours**

Classification:

Decision Tree,The Tree Induction Algorithm,Split algorithms based on information theory,Split Algorithm based on Gini Index,Decision tree Rule,

Unit V **06 Hours**

Knowledge discovery:

Introduction, KDD Process KDD process steps, Models, Integration of KDD with Database system,KDD system architecture, KDD Lifecycle,

Unit VI **06 Hours**

Clustering:

Cluster analysis, Categorization of major clustering methods such as Partitioning methods, Hierarchical methods, Density based methods, grid based methods, Model based clustering methods, clustering high dimensional data, Constraint based analysis, Data mining applications

Term Work:

1. Implementing Web document browsing a OLAP using existing ontologies.
2. Show the implementation of Naïve Bayes algorithm.
3. Demonstration of Association rule process on any dataset using apriori algorithm.
4. Case Study: How New York's Fire Department Uses Data Mining
5. Comparison of various data mining tool
6. Implementation/usage of WEKA for classification of social network dataset

7. Implementation/usage of k-nearest neighbor classifier
8. Find predominant themes in a collection of documents (clustering).
9. Clustering images based on feature localization.
10. Case Study on Data mining applications.

Assignment:

1. Explain Data mining Tasks, Issues, Evalution and Terminologies.
2. Implementation of various KDD models.
3. Define hypercube? How do they apply in OLAP system?
4. State Codd's guidelines for OLAP system?
5. Case study on OLAP.
6. Apply Apriori algorithm to the dataset from Table 13.1 (textbook) and extract

   all frequent itemsets with support count ≥3. For one longest itemset construct association rules using confidence threshold 50%. Show all your work (candidates, frequent itemsets, etc.).
7. List and explain Association rule mining algorithms in detai.
8. Implementation of Data Cube.
9. Compare various clustering algorithm.
10. Case Study :clustering algorithm

Text books:

1. Jiawei Han & Micheline Kamber, "Data Mining – Concepts and Techniques",Morgan Kaufmann Publishers, Elsevier,2nd Edition, 2006.
2. ",Pang-Ning Tan, Michael Steinbach and Vipin Kumar "Introduction to Data Mining, Pearson education.

Reference books

1. Arun K Pujari, "Data Mining Techniques",2nd edition, Universities Press.
2. Sam Aanhory & Dennis ,"Data Warehousing in the Real World",Murray Pearson Edn Asia.
3. K.P.Soman,S.Diwakar,V.Ajay, "Insight into Data Mining",PHI,2008.
4. Paulraj Ponnaiah, "Data Warehousing Fundamentals ",Wiley student Edition.

Syllabus for Unit Test:

Unit Test -1 UNIT – I, UNIT – II, UNIT - III

Unit Test -2 UNIT – IV, UNIT – V, UNIT - VI

Project Stage-II

Teaching Scheme	Examination Scheme	Credit Scheme
Practical:16 Hours / Week	End Semester Examination: 150 Marks	Term Work & Oral: 08

Course Pre-requisites:

Basics of Software engineering, software testing and knowledge of core computer engineering subjects.

Course Objectives:

- To develop problem solving abilities using mathematics.
- To apply algorithmic strategies while solving problems.
- To prepare software engineering documents and design test cases.
- To demonstrate use of algorithmic solutions in real time problem.
- To encourage and expose students for participation in National/ International paper presentation activities.
- Exposure to Learning and knowledge access techniques using Conferences, Journal papers and participation in research activities.

Course Outcomes: Upon completion of the course student will able to

I. Understand how to solve the problem.

II. Demonstrate the ability to synthesize complex information from a variety of sources in decision-making

III. Plan and perform collaboratively towards a common purpose.

IV. Demonstrate self-advocacy skills and self-reliant behavior.

V. Demonstrate the ability to develop and maintain satisfying interpersonal relationships.

VI. Evaluate and conclude the results with documentation.

1. The project will be undertaken preferably by a group of at least 3- 4 students who will jointly work and implement the project over the academic year. The work will involve the design of a system or subsystem in the area of Computer Engineering.
2. If the project is chosen a hardware project it will involve the designing a system –subsystem or upgrading an existing system. The design must be implemented into a working model with necessary software interfacing and a user manual.

3. If the project is chosen in the pure Software Application it must involve the detail Software Design Specifications, Data Structure Layout, File Design, Testing with complete documentation and user interface. With life cycle testing and as an executable package.

   The group will submit at the end of Semester-VIII,

   i) The workable project.

   ii) The details of Research paper published in National/ International paper conferences/journals for the project work carried out.

   iii) Project Report in the form of bound journal complete in all aspects, 3 copies for the institute and 1 copy of each student in the group for certification.

   The examiner in consultation with the guide will assess the term work. Oral examination will be based on the project work completed by the candidate.

   The project report will contain the following details:

   1. Problem definition and requirement specification, acceptance tests procedure (ATP).
   2. System definition, requirement analysis.
   3. System design.
   4. System implementation-code documentation –dataflow diagram / algorithm.
   5. Test results and procedure, test report as per ATP.
   6. Platform choice, use.
   7. Appendix tools used, references.

8. Documentation will use UML approach with Presentation, Category, Use Case, Class Diagrams, etc.