

**RAJEEV GANDHI MEMORIAL
COLLEGE OF ENGINEERING AND TECHNOLOGY**
AUTONOMOUS
NANDYAL-518501, KURNOOL DIST., A.P., INDIA

COMPUTER SCIENCE & ENGINEERING



II, III & IV B.Tech SYLLABUS

Applicable for students admitted into B.Tech (Regular) from 2012-13
&
B.Tech (Lateral Entry Scheme) from 2013-14

REGULATIONS, COURSE STRUCTURE & DETAILED SYLLABUS

RAJEEV GANDHI MEMORIAL COLLEGE OF ENGINEERING AND TECHNOLOGY, NANDYAL-518501, KURNOOL (DIST), A.P., INDIA

AUTONOMOUS INSTITUTE

(Affiliated to J.N.T.U.A, Anantapur)

ACADEMIC REGULATIONS, COURSE STRUCTURE AND DETAILED SYLLABI

B.Tech (Regular) from 2012-13 and B.Tech (Lateral Entry Scheme) from 2013-14

For pursuing four year under graduate Bachelor Degree Programme of study in Engineering (B.Tech), Two year Master (post graduate) Degree of study in Engineering (M.Tech), Two year Master (post graduate) degree of study in Business Administration (MBA), Three year Master (post graduate) Degree of study in Computer Applications (MCA) offered by Rajeev Gandhi Memorial College of Engineering and Technology, Nandyal - 518501 under Autonomous status and herein referred to as RGM CET (Autonomous):

All the rules specified herein approved by the Academic Council will be in force and applicable to students admitted from the Academic Year 2012-13 onwards. Any reference to “Institute” or “College” in these rules and regulations shall stand for Rajeev Gandhi Memorial College of Engineering and Technology (Autonomous).

All the rules and regulations, specified hereafter shall be read as a whole for the purpose of interpretation as and when a doubt arises, the interpretation of the Chairman, Academic Council is final. As per the requirements of statutory bodies, the Principal, Rajeev Gandhi Memorial College of Engineering and Technology shall be the Chairman, Academic Council.

The candidate seeking admission into the first year of study of four year B.Tech degree Programme should have

- i) Passed either Intermediate Public Examination (IPE) conducted by the Board of Intermediate Education, Andhra Pradesh with Mathematics, Physics and chemistry as optional subjects (or any equivalent examination certified by Board of Intermediate Examinations) or a Diploma in Engineering in the relevant branch conducted by the Board of Technical Education, Andhra Pradesh (or any equivalent certified by State Board of Technical Education) for admission.
- ii) Secured a rank in the EAMCET examination conducted by AP State Council for Higher Education (APSCHE) for allotment of a seat by the Convener, EAMCET, for admission.

Admission Procedure:

As per the norms of A.P. State Council of Higher Education (APSCHE), Government of Andhra Pradesh, admissions are made to the first year of Four year B.Tech. Degree programme as follows:-

- a) As per the norms of Government of Andhra Pradesh, A-Category (based on the rank obtained in EAMCET) seats will be filled by the Convener, EAMCET.
- b) As per the norms of Government of Andhra Pradesh, B-Category seats will be filled by the management.

Admission to the Second year of Four year B.Tech. Degree Programme in Engineering:

- i) Candidates qualified in ECET and admitted by the Convener, ECET, in such cases for admission, when needed permission from the statutory bodies is to be obtained.
- ii) 20% of the sanctioned strength in each programme of study (of RGM CET) shall be filled by the Convener, ECET as lateral entry.

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**List of Programs offered**

1. B.Tech – Regular (& Lateral Entry)
2. M.Tech – Regular
3. MBA – Regular
4. MCA – Regular

Academic Regulations for 2012 B. Tech (Regular)

(Effective for the students admitted into the I year from the Academic Year 2012-2013)

The B.Tech degree will be conferred by the Jawaharlal Nehru Technological University, Anantapur, to those students who are admitted to the program and fulfil all the requirements for the award of the Degree as specified below.

1.0 Award of B.Tech. Degree

- 1.1 The student will be declared eligible for the award of the B. Tech. degree if he fulfils the following academic regulations:
- 1.2 Pursued a course of study for not less than prescribed course work duration and not more than double the prescribed course work duration.
- 1.3 Registered for 196 credits and secured 190 credits with compulsory subjects as listed in Table-1 below.

Table 1: Compulsory Subjects

S.NO	SUBJECT PARTICULARS
1.	All the first year subjects
2.	All practical subjects
3.	Skill Development Courses
4.	Mini project
5.	Seminar
6.	Comprehensive viva voce
7.	Project work

2.0 Forfeit of seat

Students, who fail to fulfil all the academic requirements for the award of the degree within ***eight academic years*** from the year of their admission, shall forfeit their seat in B.Tech course.

3.0 Courses of study

The following courses of study are offered at present as specializations for the B. Tech. Course

1. Civil Engineering.
2. Computer Science and Engineering.
3. Electrical and Electronics Engineering.
4. Electronics and Communication Engineering.
5. Electronics and Instrumentation Engineering.
6. Information Technology.
7. Mechanical Engineering.

and any other course as approved by the authorities of the University from time to time.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Table 2: Credits

	I Year				Semester			
	Periods /Week	Credits	Internal Marks	External Marks	Periods / Week	Credits	Internal Marks	External Marks
Theory	02	02	30	70	04	03	30	70
	03	03	30	70				
	03+1*	03	30	70				
	03+1*	04 or 05	30	70				
Practical	03	03	25	50	03	02	25	50
Practical / Drawing	3+1*	02			06	03		
	06	04	30	70			30	70
Skill Development Courses	03					02**	100	
Mini Project						02		50
Seminar						02	50	
Comprehensive Viva-voce						03		50
Project						10	50	100

[*Tutorial

****Skill Development Courses credits will not be considered for the award of division. However all these courses have to be cleared through Internal evaluation by scoring minimum of 40%.The credits obtained in these courses will be taken in to account for award of degree.]**

4.0 Distribution and Weightage of Marks

- 4.1 The performance of the student in each semester / I year shall be evaluated subject – wise with a maximum of 100 marks for theory and 75 marks for practical subject. In addition, mini-project, comprehensive viva, seminar shall be evaluated for 50 marks each and the project work shall be evaluated for 150 marks.
- 4.2 For theory subjects the distribution shall be 30 marks for Internal Evaluation (25 marks for Internal test and 05 marks for assignments) and 70 marks for the End-Examination.
- 4.3 For the semester system, during the semester there shall be 2 tests for theory subjects. In each Internal test there shall be one compulsory (short answers) question and 3 descriptive questions are to be answered. The duration of Internal test will be for 2hrs. First test to be conducted in 3 units and second test to be conducted in remaining 3 units of each subject. For awarding of 25 Internal marks the performance of the student in two Internal examinations conducted one in the middle of the semester and the other towards the end of the semester giving a weightage of 0.75 for the better score and 0.25 for the other score will be considered. There shall be two assignments (problem based) in each semester for award of 05 marks so that Internal component (marks) will be 30 marks (25 marks for Internal test+05 marks for assignments).
- 4.4 For the I year class which shall be on yearly basis, there shall be 3 tests. For awarding of 25 Internal marks the performance of the student in three Internal examinations conducted as per the schedule giving a weightage of 0.5 for the best score, 0.25 for better score and 0.25 for the other score will be considered. The distribution of syllabus for the conduct of Internal tests in the first year shall be as follows:

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**Table 3: Units for Internal Tests**

I Year	Semester
2 Units First Internal test.	3 Units First Internal test.
2 Units Second Internal test.	3 Units Second Internal test.
2 Units Third Internal test.	

In a year there shall be at least three assignments and in each semester there shall be two assignments for the award of 5 marks.

- 4.5 In the case of Skill Development Courses two Internal examinations shall be conducted one in the middle of the semester and the other at the end of the semester for 70 marks and the marks scored by the student in these exams with a weightage of 0.75 for better score and 0.25 for the other score will be awarded as Internal marks for 70. The remaining 30 marks are based on the average marks scored in two assignment. No external exam for these courses.
- 4.6 No makeup test for Internal examination or assignments will be conducted in any subject or practical. The student, who is absent for any test shall be deemed to have scored zero in that test.

5.0 Question Paper Pattern:

- 5.1 Each Internal Test question paper shall contain 5 questions, of which the First question is compulsory and three questions are to be answered from the remaining four. Compulsory question carries 10 marks (It contains 5 questions of two marks - no choice in first question). The remaining 3 questions carry 5 marks each.
- 5.2 The End Examination question paper will have 7 questions and students have to write 5 questions. However, the first question is compulsory and it consists of 7 short answer questions, each carrying 2 marks. The next 4 questions are to be answered from the remaining 6 questions and each carries 14 marks.
- 5.3 For practical subjects there shall be a continuous evaluation during the semester for 25 Internal marks and 50 End Examination marks. Of the 25 marks for Internal, 20 marks shall be awarded for day-to-day work and 5 marks to be awarded by conducting an Internal laboratory test. The End Examination shall be conducted by the teacher concerned and an external Examiner from other institutions.
- 5.4 For the subject having design and / or drawing, (such as Engineering Graphics, Machine Drawing etc) and estimation, the distribution shall be 30 marks for Internal evaluation (15 marks for day-to-day work and 5 marks for Internal tests and 10 marks for assignments) and 70 marks for End Examination. There shall be two Internal tests in a Semester and the best of the two shall be considered for the award of marks for Internal tests. However in the I year class, there shall be three Internal tests and the average of best two will be taken into consideration for award of Internal marks.
- 5.5 The Engineering Drawing Practice Lab, wherever offered is to be treated as a theory subject. Evaluation method adopted for theory subjects shall be followed here as well.
- 5.6 There shall be mini-Project, in collaboration with an industry (wherever possible) of their specialization, to be taken up during the vacation (data collection, components etc) after III year II Semester examination and implementation/simulation shall be carried out in IV year first semester during lab classes. Implementation or construction of mini project will be treated as laboratory. However, the mini project and its report shall be evaluated in IV year I Semester. The mini project shall be submitted in report form and should be presented before the committee, which shall be evaluated for 50 marks. The committee consists of an external Examiner, Head of the Department, the supervisor of mini project and a senior faculty member

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

of the Department. There shall be 25 Internal marks for mini project which will be awarded based on the performance and involvement of the student during mini project period.

- 5.7 There shall be a seminar presentation in IV year II semester. For the seminar, the student shall collect the information on a specialized topic and prepare a technical report, showing his understanding over the topic, and submit to the Department, which shall be evaluated by the Departmental committee consisting of Head of the Department, seminar supervisor and a senior faculty member of the department. The seminar report shall be evaluated for 50 marks. There shall be no external examination for seminar.
- 5.8 There shall be a comprehensive viva voce examination at the end of IV year II semester for 50 marks which shall be conducted by HOD, senior faculty and external Examiner from other institute.
- 5.9 The project topic should be approved by Internal Department Committee (IDC). Out of total 150 marks for the project work, 50 marks shall be for Internal Evaluation and 100 marks for the End Semester Examination. The evaluation of project work shall be conducted at the end of the IV year II semester. The project viva voce examination will be conducted by the committee consists of an external Examiner from other institute, Head of the Department and the supervisor of the project. The Internal evaluation for 50 marks shall be on the basis of two seminars given by each student on the topic of the project. The Internal evaluation of the project work for 50 marks shall be conducted by the committee consists of head of the Department or his nominee, senior faculty member and the supervisor of project.

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**Table4: Distribution of weightages for examination and evaluation:**

S.No	Nature of subject	Marks	Type of examination and mode of assessment		Scheme of Examination
1	Theory	70	End Examination Double Evaluation (Internal+External evaluation)		End Examination in theory subjects will be for 70 marks.
		30	25	Internal examinations (Internal evaluation)	These 25 marks are awarded to the students based on the performance in three(yearly) or two(semester) Internal examinations with a weightage of 0.5 for best score ,0.25 for better score ,0.25 for other score (yearly) and 0.75 for better score and 0.25 for the other score(semester) respectively.
			05	Assignments (Internal evaluation)	Average of Three/two assignments in a year/ semester each of 05 marks.
2	Practical	50	End lab examination (External evaluation)		This End Examination in practical subjects will be for a maximum of 50 marks.
		25	20	Internal evaluation	Day-to-day performance in lab experiments and record
			05	Internal evaluation	Internal lab examination at the end of year/semester
3	Mini Project	50	End Examination (External evaluation)		This End Examination in miniproject will be for a maximum of 50 marks.
		25	Internal evaluation		Day-to-day performance in executing mini project .
4	Seminar	50	Internal evaluation		Based on the performance in two seminars during semester
5	Comprehensive Viva	50	External evaluation		This end viva voce examinations in all the subjects for 50 marks
6	Project work	100	External evaluation		This end viva voce in project work for 100 marks
		50	Internal evaluation		These 50 marks will be based on the performance of the student in the project reviews apart from attendance and regularity
7	Skill Development Courses	70	Internal evaluation		These 70 marks are awarded to the students based on the performance of two Internal examinations with a weightage of 0.75 for better score and 0.25 for the other score
		30	Internal evaluation		Based on the two assignments

6.0 Attendance Requirements:

- 6.1 The student shall be eligible to appear for End Examinations of the semester/ year if he acquires a minimum of 75% of attendance in aggregate of all the subjects of that semester/year.
- 6.2 Condonation of shortage of attendance in aggregate up to 10% (65% and above and below 75%) in a semester / year may be granted by the College Academic Committee.
- 6.3 The student will not be promoted to the next semester unless he satisfies the attendance requirement of the present semester / year, as applicable. They may seek re-admission for that semester / year when offered next.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

- 6.4 **Shortage of Attendance below 65% in aggregate shall in NO case be condoned.**
- 6.5 Students whose shortage of attendance is not condoned in any semester / year are not eligible to take their End Examination of that class and their registration shall stand cancelled.
- 6.6 The stipulated fee shall be payable towards condonation of shortage of attendance.

7.0 Minimum Academic Requirements:

The following academic requirements have to be satisfied in addition to the attendance requirements mentioned in item no.6.0.

- 7.1 The student shall be deemed to have satisfied the minimum academic requirements and earned the credits allotted to each theory or practical or design or drawing subject or Skill Development Courses or project if he secures not less than 35% of marks in the End Examination and he has to score minimum of 40% marks from Internal and external exam marks put together to clear the subject.
- 7.2 The student shall be promoted from II to III year only if he fulfils the academic requirement of securing 46 out of 92 credits from all the exams conducted upto and including II year II semester regular examinations (**Two regular and one supplementary examinations of I year; one regular and one supplementary examinations of II year I semester; one regular examination of II year II semester**) irrespective of whether the candidate takes the examination or not.
- 7.3 The student shall be promoted from third year to fourth year only if he fulfils the academic requirements of securing total 72 out of 144 credits from all the exams conducted upto and including III year II semester regular examinations ,whether the candidate takes the examinations or not. (**Three regular and two supplementary examinations of I year; Two regular and two supplementary examinations of II year I semester ; Two regular and one supplementary examinations of II year II semester ; One regular and one supplementary examination of III year I semester ; One regular examination of III year II semester**)

Table 5: Promotion rules

Promotion from	Total credits to register	Total credits to obtain for promotion
II yr to III yr	92	46
III yr to IV yr	144	72

- 7.4 The student shall register and put up minimum attendance in all 196 credits and earn the 190credits. Marks obtained in the best 178 credits (excluding the credits obtained in Skill Development Courses) shall be considered for the calculation of percentage of marks.
- 7.5 Students who fail to earn 190 credits as indicated in the course structure including compulsory subjects as indicated in Table-1 within eight academic years from the year of their admission shall forfeit their seat in B.Tech course and their admission shall stand cancelled.

8.0 Course pattern:

- 8.1 The entire course of study is of four academic years. The first year shall be on yearly pattern and the second, third and fourth years shall be on semester pattern.
- 8.2 The student is eligible to appear for the End Examination in a subject, but absent at it or has failed in the End Examination may appear for that subject at the supplementary examination.

AUTONOMOUS
SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Table 6: Course pattern

Year	Semester	No. of Subjects	No. of Skill Development Courses	Number of Labs	Total credits
First year		O7 {ENG-3 EP-4, EC-4, M1-4, MM/EM-4, CP-5,ED-4}	00	04	1X3=03 4X5=20 5X1=05 4X3=12 40
Second year	First	06	01	03	6X3=18 1X2=02 3x2=06 26
	Second	06	01	03	6X3=18 1X2=02 3x2=06 26
Third year	First	06	01	03	6X3=18 1X2=02 3x2=06 26
	Second	06	01	03	6X3=18 1X2=02 3x2=06 26
Fourth year	First	06	01	02 Mini project	6X3=18 1X2=02 3x2=06 26
	Second	03	01	Subjects Open elective Seminar Comprehensive Viva Project Viva	3x3 =09 1X2=02 1X2=02 1X3=03 1X10=10 26
GRAND TOTAL					196

9.0 Transitory Regulations:

Candidates who have been detained for want of attendance or not fulfilled academic requirements or who have failed after having undergone this course in earlier regulations or have discontinued and wish to continue the course are eligible for admission into the unfinished semester from the date of commencement of class work with the same or equivalent subjects as and when subjects are offered subject to section 2.0 and they continues to be in the academic regulations they were first admitted.

10.0 With-holding of results:

If the candidate has any dues not paid to the Institute or if any case of indiscipline of malpractice is pending against him, the result of the candidate shall be withheld and he will not be allowed/promoted into the next higher semester. The issue of degree is liable to be withheld in such cases.

11.0 Award of Class:

After the student has satisfied the requirements prescribed for the completion of the programme and is eligible for the award of B. Tech. degree he shall be placed in one of the following four classes:

Table 7: Award of Division

Class Awarded	% of marks to be secured	From the aggregate marks secured for the best 178 Credits (excluding Skill Development Courses)
First Class with Distinction	70% and above	
First Class	Below 70% but not less than 60%	
Second Class	Below 60% but not less than 50%	
Pass Class	Below 50% but not less than 40%	

(The marks in Internal evaluation and End Examination shall be shown separately in the marks memorandum)

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**11.0 Supplementary Examinations:**

Apart from the regular End Examinations the institute may also schedule and conduct supplementary examinations for all subjects for the benefit of students with backlogs. Such students writing supplementary examinations as supplementary candidates may have to write more than one examination per day.

12.0 Transcripts:

After successful completion of prerequisite credits for the award of degree a Transcript containing performance of all academic years will be issued as a final record. Duplicate transcripts will also be issued if required after the payment of requisite fee and also as per norms in vogue.

13.0 Rules of Discipline:

- 12.1 Any attempt by any student to influence the teachers, Examiners, faculty and staff of controller of Examination for undue favours in the exams, and bribing them either for marks or attendance will be treated as malpractice cases and the student can be debarred from the college.
- 12.2 When the student absents himself, he is treated as to have appeared and obtained zero marks in that subject(s) and grading is done accordingly.
- 12.3 When the performance of the student in any subject(s) is cancelled as a punishment for indiscipline, he is awarded zero marks in that subject(s).
- 12.4 When the student's answer book is confiscated for any kind of attempted or suspected malpractice the decision of the Examiner is final.

14.0 Minimum Instruction Days:

The minimum instruction days for each semester / I year shall be 95/180 clear instruction days excluding the days allotted for tests/examinations and preparation holidays declared if any.

15.0 Amendment of Regulations:

The college may, from time to time, revise, amend or change the regulations, scheme of examinations and syllabi. However the academic regulations of any student will be same throughout the course of study in which the student has been admitted.

16.0 Transfers

There shall be no branch transfers after the completion of admission process.

17.0 General:

- 16.2 The Academic Regulation should be read as a whole for the purpose of any interpretation.
- 16.2 In the case of any doubt or ambiguity in the interpretation of the above rules, the decision of the Academic Council is final.
- 16.3 The Institute may change or amend the academic regulations or syllabi at any time and the changes or amendments made shall be applicable to all the students with effect from the dates notified by the Institute.
- 16.4 Where the words "he", "him", "his", occur in the regulations, they include "she", "her", "hers".

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**Academic Regulations for B. Tech.
(Lateral Entry Scheme)**

(Effective for the students getting admitted into II year from the Academic Year 2013-2014 onwards)

- 1.0 The Students have to acquire 150 credits out of 156 from II to IV year of B.Tech. Program (Regular) for the award of the degree.
- 2.0 Students, who fail to fulfil the requirement for the award of the degree in 6 consecutive academic years from the year of admission, shall forfeit their seat.
- 3.0 The same attendance regulations are to be adopted as that of B. Tech. (Regular).

4.0 Promotion Rule:

The student shall be promoted from third year to fourth year only if he fulfils the academic requirements of 52 out of 104 credits from all the exams conducted upto and including III year II semester regular examinations, whether the candidate takes the examinations or not. **(Two regular and Two supplementary examinations of II year I semester; Two regular and one supplementary examinations of II year II semester; One regular and one supplementary examination of III year I semester; One regular examination of III year II semester).**

5.0 Award of Class:

After the student has satisfied the requirements prescribed for the completion of the program and is eligible for the award of B. Tech. Degree he shall be placed in one of the following four classes: The marks obtained in the best 138 credits will be considered for the calculation of percentage and award of class.

Table 1: Award of Division

Class Awarded	% of marks to be secured	From the aggregate marks secured for best 138 Credits. (i.e. II year to IV year) excluding Skill Development Courses
First Class with Distinction	70% and above	
First Class	Below 70% but not less than 60%	
Second Class	Below 60% but not less than 50%	
Pass Class	Below 50% but not less than 40%	

(The marks in Internal evaluation and End Examination shall be shown separately in the marks memorandum)

- 6.0 All other regulations as applicable for B. Tech. Four-year degree course (Regular) will hold good for B. Tech. (Lateral Entry Scheme)

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**I B.Tech****COURSE STRUCTURE**

(Common to Branches: ECE, EEE, EIE, CSE & IT)

Code	Subject	Scheme of instruction periods/week		Credits	Scheme of Examination		
		Theory	Practical		Internal Marks	External Marks	Total Marks
Theory							
A0001121	Professional English	3+1*	-	3	30	70	100
A0002121	Engineering Physics	3+1*	-	4	30	70	100
A0003121	Engineering Chemistry	3+1*	-	4	30	70	100
A0004121	Mathematics – I	3+1*	-	4	30	70	100
A0005121	Mathematical Methods	3+1*	-	4	30	70	100
A0501121	Fundamentals of Computers & C Programming	3+1*	-	5	30	70	100
A0301121	Engineering Drawing	6	-	4	30	70	100
Practical							
A0591121	Computer Programming Lab	-	3	3	25	50	75
A0391121	Engineering and IT Workshop	-	3	3	25	50	75
A0091121	Engineering Physics Lab and Engineering Chemistry Lab	-	3	3	25	50	75
A0092121	English Language Communication Skills Lab	-	3	3	25	50	75
Total		30	12	40	310	690	1000

AUTONOMOUS
SCHOOL OF COMPUTER SCIENCE AND ENGINEERING
II B.TECH, I-SEMESTER COURSE STRUCTURE

Subject Code	Subject	Hours/ Week			Credits	Marks		
		Theory	Tutorial	Lab		Internal	External	Total
Theory								
A0012123	Probability and Statistics	3	1	-	3	30	70	100
A0502123	Digital Logic Design	3	1	-	3	30	70	100
A0503123	Mathematical Foundations of Computer Science	3	1	-	3	30	70	100
A0504123	Data Structures	3	1	-	3	30	70	100
A0010123	Environmental Studies	3	1	-	3	30	70	100
A0206123	Principles of Electrical and Electronics Engineering	3	1	-	3	30	70	100
A0009123	Corporate Management Skills(Skill Development Course)	3	-	-	2	30+70	-	100
Practical								
A0592123	Data Structures Lab	-	-	3	2	25	50	75
A0593123	Digital Logic Design Lab	-	-	3	2	25	50	75
A0294123	Principles of Electrical and Electronics Engineering Lab	-	-	3	2	25	50	75
Contact Periods / Week		21	6	9	26	355	570	925

II B.TECH, II-SEMESTER COURSE STRUCTURE

Subject Code	Subject	Hours/ Week			Credits	Marks		
		Theory	Tutorial	Lab		Internal	External	Total
Theory								
A0011123	Managerial Economics & Financial Analysis	3	1	-	3	30	70	100
A0505124	Computer System Design	3	1	-	3	30	70	100
A0506124	Object Oriented Programming	3	1	-	3	30	70	100
A0507124	Formal Languages and Automata Theory	3	1	-	3	30	70	100
A0508124	Database Management Systems	3	1	-	3	30	70	100
A0509124	Operating Systems	3	1	-	3	30	70	100
A0007123	Aptitude Arithmetic Reasoning & Comprehension (Skill Development Course)	3	-	-	2	30+70	-	100
Practical								
A0594124	Operating Systems Lab	-	-	3	2	25	50	75
A0595124	Object Oriented Programming Lab	-	-	3	2	25	50	75
A0596124	Database Management Systems Lab	-	-	3	2	25	50	75
Contact Periods / Week		21	6	9	26	355	570	925

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

III B.TECH, I-SEMESTER COURSE STRUCTURE

Subject Code	Subject	Hours/ Week			Credits	Marks		
		Theory	Tutorial	Lab		Internal	External	Total
Theory								
A0510125	Advanced Java Programming	3	1	-	3	30	70	100
A0511125	Computer Networks	3	1	-	3	30	70	100
A0512125	Principles of Programming Languages	3	1	-	3	30	70	100
A0513125	Compiler Design	3	1	-	3	30	70	100
A0514125	Software Engineering	3	1	-	3	30	70	100
A0515125	C# and .Net Framework	3	1	-	3	30	70	100
A0013125	Professional Ethics and soft skills(Skill Development Course)	3	-	-	2	30+70	-	100
Practical								
A0597125	Advanced Java Programming Lab	-	-	3	2	25	50	75
A0598125	C# and .Net Framework Lab	-	-	3	2	25	50	75
A0599125	Computer Networks & Compiler Design Lab	-	-	3	2	25	50	75
	Contact Periods / Week	21	6	9	26	355	570	925

III B.TECH, II-SEMESTER COURSE STRUCTURE

Subject Code	Subject	Hours/ Week			Credits	Marks		
		Theory	Tutorial	Lab		Internal	External	Total
Theory								
A0517126	Mobile Computing	3	1	-	3	30	70	100
A0518126	Design And Analysis Of Algorithms	3	1	-	3	30	70	100
A1229126	Information Security	3	1	-	3	30	70	100
A0520126	Software Testing Methodologies And Tools	3	1	-	3	30	70	100
A0521126	UNIX and Shell Programming	3	1	-	3	30	70	100
	<u>Elective-I</u>	3	1	-	3	30	70	100
A0526126	Introduction to Oracle 9i (Skill Development Course)	3	-	-	2	30+70	-	100
Practical								
A0581126	Software Testing Tools Lab	-	-	3	2	25	50	75
A1294124	UNIX and Shell Programming Lab	-	-	3	2	25	50	75
A0582126	Design and Analysis of Algorithms Lab	-	-	3	2	25	50	75
	Contact Periods / Week	21	6	9	26	355	570	925

AUTONOMOUS
SCHOOL OF COMPUTER SCIENCE AND ENGINEERING
IV B.TECH, I-SEMESTER COURSE STRUCTURE

Subject Code	Subject	Hours/ Week			Credits	Marks		
		Theory	Tutorial	Lab		Internal	External	Total
Theory								
A0528127	Computer Graphics	3	1	-	3	30	70	100
A0527127	Web Technologies & Programming	3	1	-	3	30	70	100
A0529127	Data Warehousing & Mining	3	1	-	3	30	70	100
A0530127	Software Project Management	3	1	-	3	30	70	100
	Elective-II	3	1	-	3	30	70	100
	Elective-III	3	1	-	3	30	70	100
A0537127	CCNA (Skill Development Course)	3	-	-	2	30+70	-	100
Practical								
	Elective-II Lab	-	-	3	2	25	50	75
A0586127	Web Technologies & Programming Lab	-	-	3	2	25	50	75
A0587127	Mini Project	-	-	3	2	25	50	75
	Contact Periods / Week	21	6	9	26	355	570	925

IV B.TECH, II-SEMESTER COURSE STRUCTURE

Subject Code	Subject	Hours/ Week			Credits	Marks		
		Theory	Tutorial	Lab		Internal	External	Total
Theory								
A0014125	Management Science	3	1	-	3	30	70	100
	Elective-IV	3	1	-	3	30	70	100
	Elective-V	3	1	-	3	30	70	100
A0015125	Man Management (Skill Development Course)	3	-	-	2	30+70	-	100
Project, Seminar, Comprehensive Viva								
A0588128	Seminar	-	-	-	2	50	-	50
A0589128	Comprehensive Viva-Voce	-	-	-	3	-	50	50
A0590128	Project Work	-	-	-	10	50	100	150
	Contact Periods / Week	12	3	-	26	290	360	650

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**ELECTIVES**

S.No	Code	Name of the Subjects
Elective-I		
1	A0522126	Cloud Computing
2	A0523126	Distributed Systems
3	A0524126	Advanced Computer Architecture
4	A0525126	Virtual Reality
Elective-II		
1	A0531127	SAP ABAP and Basic Applications
2	A0532127	Network Programming
3	A0423127	Digital Image Processing
4	A1220126	Embedded Computing
Elective-III		
1	A0533127	Distributed Databases
2	A0534127	Fundamentals of Soft Computing
3	A0535127	Object Oriented Software Engineering.
4	A0536127	Enterprise Application Integration
Elective-IV (Open Elective)		
1	A1227126	Human Computer Interaction
2	A0538128	Intellectual Property Rights & Cyber Law
3	A0539128	Green IT
4	A0540128	Computer Forensics
Elective-V		
1	A0541128	Concepts of Machine Learning
2	A0542128	Game Theory
3	A1225126	Multimedia and Application Development
4	A0543128	Adhoc & Sensor Networks
Elective-II Lab		
1	A0583127	SAP ABAP Applications Lab
2	A0584127	Network Programming Lab
3	A0585127	Digital Image Processing Lab
4	A1298126	Embedded Computing Lab

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

II B.Tech. I-Sem (CSE)

T C
3+1* 3**(A0012123) PROBABILITY AND STATISTICS**

(Common to CSE, IT, ME &CE)

OBJECTIVES:

- Probability & Statistics is a necessary avenue to scientific knowledge which opens new vistas of mental activity.
- There was a great need for an associate knowledge on Probability & Statistics for the engineering students.
- It is accepted that a good mathematical studying is essential for all engineers; this will definitely boost the confidence of the student in writing “Competitive examinations”.

OUTCOMES:

By the end of module students will be expected to demonstrate knowledge of

- Probability, Conditional Probability, Baye’s theorem and its applications.
- Random variables, Discrete random variables, Continuous random variables.
- Binomial Distribution, Poisson Distribution, Normal Distribution.
- Population & Samples, Sampling Distribution of means.
- Point Estimation, interval Estimation, Bayesian Estimation.
- Tests of Hypothesis, Z – Distribution, Student t – test, F – test, Chi square test.
- Queuing theory, Basic Queuing process, Transient & Steady states, Pure birth & Death process.

UNIT – I

Probability: Sample Space and events – Probability – The axioms of Probability – Some Elementary theorems – Conditional Probability – Baye’s theorem.

UNIT – II

Random Variables: Discrete and continuous - Distribution – Distribution functions – Properties – Discrete Random variables – Probability mass function – Continuous Random variables – Probability density function.

UNIT – III

Binomial, Poisson and Normal distributions – Related properties – Fitting distributions.

UNIT – IVSampling distribution : Population and samples – Sampling distribution of mean (known and unknown)
Estimation: Point estimation – Interval estimation – Bayesian estimation.**UNIT – V**

Test of Hypothesis – Means – Proportions – Hypothesis concerning one and two means – One tail, two tail tests – Type I and Type II errors.

UNIT – VITests of significance – Student’s t –test, F – test, Ψ^2 test – Good ness of fit – Contingency test.**TEXT BOOKS:**

- 1) Probability and Statistics by T.K.V. Iyengar, B. Krishna Gandhi and others, S.Chand and company.
- 2) A Text book of Probability and Statistics by Dr. Shahnaz Bathul.
- 3) Engineering Mathematics by B.V. Ramana, Tata McGraw Hill .

REFERENCES:

- 1) Fundamental Mathematical Statistics by S.C. Gupta and V.K. Kapoor – S. Chand Co.
- 2) A text book of Engineering Mathematics by N.P. Bali, Iyengar – Lakshmi Publications (Pvt ltd)
- 3) Engineering Mathematics – III A by Dr.M.K. Venkat araman – The National Publishing co.

(A0502123) DIGITAL LOGIC DESIGN
(Common to CSE & IT)**OBJECTIVES:**

- Understand the different number system, its conversions and binary arithmetic.
- Know the fundamentals of Boolean algebra and theorems, Karnaugh maps including the minimization of logic functions to SOP or POS form.
- Analysis of logic circuits and optimization techniques to minimize gate count, signals, IC count, or time delay.
- To strengthen the principles of logic design and use of simple memory devices, flip-flops, and sequential circuits.
- To fortify the documentation standards for logic designs, standard sequential devices, including counters and registers.
- To understand design of logic functions using PLDs (ROM, RAM, PAL, PLA).

OUTCOMES:

- Ability to differentiate between analog and digital representations.
- Ability to convert a number from one number system to its equivalent in of the other number system.
- Cite the advantages of the octal and hexa decimal number systems and to understand the difference between BCD and straight binary.
- Ability to perform the three basic logic operations and construct the truth tables for the different types of gates. And Implement logic circuits using basic AND, OR and NOT gates.
- Ability to Use De-Morgan's theorem to simplify logic expressions and describe the concept of active LOW and active HIGH logic signals and Use Boolean algebra and K-map as tool to simplify and design logic circuits and Design simple logic circuits without the help of truth tables.
- Ability to Construct and analyse the operation of flip-flop and troubleshoot various types of flip-flop circuits

UNIT-I

BINARY SYSTEMS: Digital Systems, Binary Numbers, Number base conversions, Octal and Hexadecimal Numbers, complements, Signed binary numbers, Binary codes, Binary Storage and Registers, Binary logic.

UNIT-II

BOOLEAN ALGEBRA AND LOGIC GATES: Basic Definitions, Axiomatic definition of Boolean Algebra, Basic theorems and properties of Boolean algebra, Boolean functions canonical and standard forms, other logic operations, Digital logic gates, and their integrated circuit numbers.

UNIT-III

GATE – LEVEL MINIMIZATION: The map method, Four-variable map, Five-Variable map, product of sums simplification Don't-care conditions, NAND and NOR implementation other Two-level implementations, Exclusive – OR function,

UNIT - IV

COMBINATIONAL LOGIC: Combinational Circuits, Analysis procedure Design procedure, Binary Adder-Subtractor, Decimal Adder, Binary Multiplier, Magnitude Comparator, Decoders, Encoders, Multiplexers, Introduction to HDL, VHDL code for basic and universal logic gates, Half adder ,full adder circuits.

UNIT - V

SYNCHRONOUS SEQUENTIAL LOGIC: Sequential circuits, latches, Flip-Flops Analysis of clocked sequential circuits, State Reduction and Assignment, Design Procedure.

ASYNCHRONOUS SEQUENTIAL LOGIC: Introduction, Analysis Procedure, Circuits with Latches, Design Procedure.

UNIT - VI

Registers, shift Registers, Ripple counters synchronous counters, other counters, , Introduction, Random-Access Memory, Memory Decoding, Error Detection and correction Read-only memory, Programmable logic Array programmable Array logic, Sequential Programmable Devices. VHDL code for Flip-flops, decade counter.

TEXT BOOKS:

- 1) Digital Design – Third Edition, M.Morris Mano, Pearson Education/PHI.
- 2) Fundamentals of Logic Design, Roth, 5th Edition, Thomson.

REFERENCES:

- 1) Switching and Finite Automata Theory by Zvi. Kohavi, Tata McGraw Hill.
- 2) Switching and Logic Design, C.V.S. Rao, Pearson Education
- 3) Digital Principles and Design – Donald D.Givone, Tata McGraw Hill, Edition.
- 4) Fundamentals of Digital Logic & Micro Computer Design , 5TH Edition, M. Rafiqzaman John Wiley.

(A0503123) MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE**OBJECTIVES:**

- To teach students notations used in the discrete mathematics associated with computer science and engineering.
- To teach the rudiments of elementary mathematical reasoning (elementary proofs; proofs by induction).
- To prepare students for the theoretical parts of all further courses in CSE.
- To study logic and Boolean algebra from a mathematical perspective, but relating it to computer engineering applications.
- To introduce basic set-theoretical notions: relations, functions, graphs, equivalence relations and orderings.
- To relate these notions to applications in CSE.

OUTCOMES:

- Understand truth tables, the concept of logical equivalence and its relationship to equivalent logic circuits and Boolean functions. Know some Boolean laws of equivalence. Extend this to predicate calculus and in predicate calculus using quantifiers.
- Be able to express English assertions in propositional calculus and in predicate calculus using quantifiers.
- Understand and use the basics of set theory notation, Boolean operations on sets Be able to work with functions.
- Be able to carry out simple direct and indirect proofs about domains like the integers and the real numbers, using quantified statements about these domains. Be able to do simple proofs by mathematical induction.
- Be able to understand and write recursive definitions, in mathematical form.
- Understand binary and n-ary relations and their applications. Know the major types of binary relations on a set. Be able to use graphs as representing relations, algorithms for relations based on graphs or matrices (e.g. transitive closure).
- Know the properties of equivalence relations and partial orderings.
- Understand lattices and Boolean algebras as universal algebras.
- Know the fundamentals of counting

UNIT-I

Mathematical Logic: Statements and notations, Connectives, Well-formed formulas, Truth Tables, tautology, equivalence, implication, Normal forms.

UNIT-II

Predicates: Rules of inference, Consistency, Predicate calculus: Free and bounded variable, Quantifiers: Universal Quantifiers, Existential Quantifiers.

UNIT-III

Relations: Relations, Properties of binary Relations, Types of relations: equivalence, compatibility and partial ordering relations, Hasse diagram. Lattices and its Properties.

UNIT-IV

Algebraic structures: Algebraic systems with examples and general properties, semi groups and monoids, groups & its types, Introduction to homomorphism and Isomorphism (Proof of theorems are not required)

Elementary Combinatory: Basis of counting, Combinations & Permutations, Permutation with repetitions, Constrained repetitions.

UNIT-V

Graph Theory: Representation of Graph, DFS, BFS, Spanning Trees, planar Graphs.

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**Unit-VI**

Graph Theory and Applications: Basic Concepts of Isomorphism and Sub graphs, Multi graphs and Euler circuits, Hamiltonian graphs, Euler's formula & its applications, Chromatic Numbers.

TEXT BOOKS:

1. Discrete Mathematical Structures with applications to computer science Trembly J.P. & Manohar .P, TMH.(**unit-1 to 4(Algebraic systems)**)
2. Discrete Mathematics for Computer Scientists & Mathematicians, J.L. Mott, A. Kandel, T.P. Baker Prentice Hall.(**unit-4(Elementary Combinatory) to 6**)

REFERENCES:

1. Mathematical foundations of computer science Dr D.S.Chandrasekharaiaha Prism books Pvt Ltd.

(A0504123) DATA STRUCTURES**OBJECTIVES:**

- This course explores fundamental data structures, algorithms for manipulating them, and the practical problems of implementing those structures in real programming languages and environments

OUTCOMES:

After the successful completion of this course, the students should:

- Understand and implement the linear data structures and implement the Non-linear data structures.

UNIT-I

Data Structures: Overview of Data Structures, **Stacks** – Introduction, Definition, Array representation of Stack, Operations on Stacks; Applications of Stacks – Infix, Prefix and Postfix Expressions; Conversions of Expressions – Infix to Prefix, Infix to Postfix; Evaluation of Postfix Expression.

UNIT-II

Queues: Introduction, Definition, Array Representation of Queue, Various Queue Structures – Ordinary Queue, Circular Queue, DEQUE, Priority Queue; Operations on Various Queue Structures – Insertion, Deletion, Search and Display, Application of Queues.

UNIT-III

Linked List: Introduction, Definition, Linked List Representation, Types of Linked Lists – Singly Linked List, Double Linked List and Circular Linked List, Operations on Linked Lists – Insertion, Deletion, Search, Display and Reverse the List, Applications of Linked Lists.

UNIT-IV

Trees: Definition and Basic Terminologies, Binary Tree, Representation of Binary Tree, Operations on a Binary Tree, Types of Binary Trees, Binary Tree traversals – In-order, Pre-order, Post-order; **Binary Search Trees:** Introduction, Definition, Representation of BST, Operations on BST.

AVL Trees: Introduction, Definition, Operations on AVL Tree – Insertion and Deletion. (Only with examples, no programmes).

UNIT-V

Graphs: Introduction – Definition, Basic Terminologies – Graph, Digraph, Weighted graph, Complete graph, Acyclic graph, Adjacent vertices, Degree of vertex, Connected graph, Representation of Graphs – Set, Linked and Sequential(matrix) representations; Operations on Graphs – Insertion, Deletion, Traversal; Graph Traversals – BFS, DFS; Applications of Graphs (only with examples no algorithms): Shortest Path Problem, Minimum Cost Spanning tree, Euler's and Hamiltonian Circuits.

UNIT-VI

Hash Tables: Hashing Techniques – Division method, Mid Square method, Folding method; Collision Resolution Techniques – Open Hashing, Linear Probing (Closed Hashing), Quadratic Probing, Double Hashing, Rehashing, External Hashing and Comparison of Collision Resolution Techniques.

TEXT BOOKS:

1. Classic Data Structures, Debasis Samanta, PHI Learning Pvt Ltd, 2nd edition.
2. An Introduction to Data Structures with Applications, Jean-Paul Tremblay and Paul G.Sorenson, Tata McGraw Hill, 2nd edition.

REFERENCES :

1. Data structures and Algorithm, Gav Pai, Tata McGraw Hill, 2nd edition.
2. Data structures Through C, G.S.Baluja, Dhanpat Rai & Co.
3. Data structures, Algorithms and Applications in C++, S.Sahni, University Press (India) pvtLtd, 2nd edition, Universities Press Orient Longman Pvt. Ltd.
4. Data structures and Algorithm Analysis in C, Mark Allen Weiss, Pearson Education. Ltd., Second Edition.

(A0010123) ENVIRONMENTAL STUDIES

(Common to all branches: E.C.E, E.E.E, E.I.E, C.S.E, I.T, M.E, C.E)

OBJECTIVES :

- To create a awareness about environment among the students.
- To develop an understanding of ecosystem and their interrelations.
- To develop an awareness about the utilization, over exploitation of natural resources.
- To recognize the need for keeping pollution under control in order to maintain the quality of life.
- To acquire skills to analyze and interpret information relating to environmental problems.
- To develop the ability to identify, analyze and reflect upon different environmental Concerns.
- To develop skills for effectively tackling problems related to the local environment.
- To adopt practices that help in promoting balance in nature by making judicious utilization of resources and materials.
- To develop love, affection, sensitivity and sense of responsibility towards all living beings.
- To appreciate and respect legal provisions for protection of animals and plants.
- To imbibe the essence of environmental values and ethics in order to live in harmony with nature.

UNIT I

MULTIDISCIPLINARY NATURE OF ENVIRONMENTAL STUDIES: Environment -Definition, scope and importance, Segments of Environment-Importance, Productivity, Aesthetical & Optional values of nature, need for public awareness.

UNIT II**HARNESSING RESOURCES**

- a) Sources of Energy:- Renewable and non-renewable resources.
- b) Natural Resources: soil, water sources-Hydro power project-problems, forest, minerals -Utilization-problems.
- c) Solar Energy and its applications - Photo Voltaic Cells, Solar water heating, solar pond, Solar Cooker. Non-conventional sources of energy.
- d) Chemical fertilizers and pesticides-problems. Green Revolution-white revolution- blue revolution.
- e) Depletion of Resources-Over utilization and consumption, non –equitable distribution of resources, Technological and Industrial developmental activities.

UNIT III

CONCEPTS OF ECO-SYSTEM : Structure of ecosystem: Tropic structure, producers, consumers, and decomposers; Interaction between biotic and a biotic factors in an ecosystem; Energy flow and its importance; Tropic levels, food chain, Food web, Food Pyramid;

TYPES OF ECOSYSTEM: Understanding the types of ecosystem

- i Terrestrial (forest, grassland and desert) and
- ii Aquatic (fresh water - River, pond and salt water-Marine) with an example of each.

UNIT IV**ENVIRONMENTAL FACTORS**

- a) Disasters:- Natural and man-made Nuclear Disasters, major types and their causes, impact on environment and human life and remedies.
- b) Impact of environment degradation on: - Natural habitats, living forms (endangered and Extinct species).
- c) Pollution:- Definition, types (soil, water, air and noise), sources , impact on physical environment control and preventive measures of pollution.

UNIT V**ENVIRONMENTAL VALUES:**

- a) Population and Environment:- Definition of species, community, population; Population. growth rate curves, Sex ratio, From unsustainable to sustainable development, Diseases-HIV, Malaria, Diaharia, Cancer.
- b) Human rights, fundamental duties and value education.
- c) Women and child welfare & Family welfare programs.

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**UNIT VI****ISSUES OF THE ENVIRONMENT**

- a) Resettlement and rehabilitation of people.
- b) Energy Crisis – urban and rural sectors.
- c) Climatic changes Greenhouse effect and global warming..
- d) Acid rain& Ozone layer depletion.
- e) Wild-life management - National parks, sanctuaries and bio-reserves, poaching, hunting and bio-piracy.
- f) E Waste Management

REFERENCES:

1. Environmental Studies by ERACH BHARUCHA for UG courses by UGC.
2. Environmental Science by Anubha Koushik & C.P Koushik, New Age International Publishers.
3. Environmental Engineering & Management by Dr.Suresh K.Dhameja, Katson books.
4. Environmental Studies by Rajagopalan, Oxford University press.
5. Environmental Studies by Manoj Tiwari & Archana Tiwari , J.K.International Publishers.
6. Environmental Studies by Benny joseph.
7. Environmental Science & Technology by M.Anji Reddy ,BS Publications.

(A0206123) PRINCIPLES OF ELECTRICAL AND ELECTRONICS ENGINEERING**OBJECTIVE:**

- This course introduces the basic concepts electric circuits and network theorems.
- This course also provides an overview of the principle, operation and applications of Diode and transistor.

OUTCOMES:

- It helps to study the basic concepts of network theorems and electronics devices.

UNIT - I

Introduction to Electrical Engineering: Essence of electricity- Conductors-Semiconductors and Insulators (Elementary treatment only); Electric field- Electric current-Potential and Potential difference, Electromotive force- Electric power-Electric Energy-Basic circuit components- Types of Sources -Ohm's law- Simple problems.

UNIT-II

Network Analysis: Basic definitions- Resistive networks- Series & Parallel circuits- Kirchoff's Laws and numerical problems, Star -Delta and Delta-Star transformation-Simple numerical problems

UNIT-III

Network Theorems: Super Position Theorem-Thevinin's Theorem-Maximum power transfer theorem- Simple problem on theorems.

UNIT-IV

Alternating Quantities : Principle of ac voltages-Waveforms and basic definitions-Root mean square and Average values of alternating currents and voltage- Form factor and Peak factor-Phasor representation of alternating quantities-Analysis of ac circuits with single basic network element.

UNIT-V

Formation of N-Type & P-type semi conductors-Construction P-N Junction diode-V-I characteristics of P-N diode-Diode Application-Half Wave Rectifier-Bridge Rectifier(Operation Only)

UNIT VI

Formation of PNP &NPN transistors-CB, CC, CE Configurations of Transistor (elementary idea only) &its applications, Application of transistors- as a Switch & as a Amplifier

TEXT BOOKS:

1. Basic Electrical Engineering –By T.K.Nagasarkar and M.S. Sukhija Oxford University Press.
2. Theory and Problems of Basic Electrical Engineering by D.P.Kothari& I.J. Nagrath PHI.

REFERENCES:

1. Principles of Electrical Engineering by V.K Mehta, S.Chand Publications.
2. Essentials of Electrical and Computer Engineering by David V. Kerns, JR. J. David Irwin Pearson.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

II B.Tech. I-Sem (CSE)

T	C
3	2

(A0009123) CORPORATE MANAGEMENT SKILLS

(Skill Development Course)

(Common to all Branches)

OBJECTIVES:

- To improve the communication skills of the students.
- To raise the confidence of the students with respect to the inter-personal communication.
- To make them to habituate to the Team culture and Team Work.
- To ensure the students to take up the challenges of Group Discussion and Personal Interview.
- To improve the overall personality of the students.

OUTCOMES:

- Able to improve the communication skills.
- Able to obtain the confidence of students with respect to the inter-personal communication.
- Able to cultivate the Team culture and Team Work.
- Able to take the challenges of Group Discussion and Personal Interview.

UNIT I

Concept of Communication – Significance, Scope and functions of Business Communication – Process and dimensions of communication – Essentials of good communication – Channels of communication – Formal, informal communication – Upward, Downward, Horizontal communication – Grapevine Phenomenon.

UNIT II

Types of communication: Verbal – Oral Communication: Advantages and limitations of oral communication, written communication – Characteristics, significance, advantages & Limitations of written communication.

UNIT III

Nonverbal Communication: Sign language – Body language – Kinesics – Proxemics – Time language and Haptics: Touch language.

UNIT IV

Interpersonal communication – Interpersonal communication – Communication models: Exchange theory – Johari window – Transactional analysis, Communication styles.

UNIT V

Managing Motivation to Influence Interpersonal communication – Inter-personal perception – Role of emotion in inter personal communication.

UNIT VI

Barriers to communication: Types of barriers – Technological – Socio-Psychological barriers – Overcoming barriers. Listening – Types of listening – Tips for effective listening..

REFERENCES:

1. Business Communication, Meenakshi Raman, Oxford University Press.
2. Business Communication, Raymond V.Lesikar, Neeraja Pandit et al., TMH
3. English for Business Communication, Dr.T.M Farhatulla, Prism books Pvt. Ltd.
4. Business Communications, Hudson, 5/e, Jaico Publications
5. Business communication for managers, Penrose, Rasberry, Myers, Cengage
6. The Skills of Communication, Bills Scot, Gower publishing company Limited, London.
7. Effective Communication, Harward Business School, Harward Business Review No.1214.
8. Essentials of Business Communication, Rajendra Pal, JS.Korlahhi, S.Chand

(A0592123) DATA STRUCTURES LAB**OBJECTIVES:**

- Learn how to implement some useful data structures.
- Understand the effect of data structures on an algorithm's complexity.
- To develop skills to design and analyze simple linear data structures
- To Strengthen the ability to identify and apply the suitable data structure for the given real world problem
- To Gain knowledge in practical applications of data structures

OUTCOMES:

- Basic ability to analyze algorithms and to determine algorithm correctness and time efficiency class.
- At the end of this lab session, the student will be able to design and analyze the time and space efficiency of the data structure
- Have practical knowledge on the application of data structures

WEEK 1:

1. Write C programs to implement the following using an array representation
 - a) Stack
 - b) Queue

WEEK 2:

2. Write C programs to implement the following using a linked representation
 - a) Stack
 - b) Queue

WEEK 3:

3. Write C program that uses Stack operations to perform the following:
 - a) Infix to Prefix
 - b) Infix to Postfix
 - c) Evaluation of Postfix

WEEK 4:

4. Write a C program that implement Circular and DE Queue with array representation.

WEEK 5:

5. Write C programs to implement the following using an array representation
 - a) Min Heap
 - b) Max Heap

WEEK 6:

6. Write C programs to implement the following using an array representation
 - a) Ascending Priority Queue
 - b) Descending Priority Queue

WEEK 7:

7. Write a C programs to implement Singly Linked List and Doubly Linked List for the following operations:
 - a) Insertion
 - b) Deletion
 - c) Search
 - d) Reverse
 - e) Display

WEEK 8:

8. Write a C program to implement Binary Tree with the following operations:
 - a) Insert an element into a binary tree.
 - b) Delete an element from a binary tree.
 - c) Search for a key element in a binary tree.

WEEK 9:

9. Write a C program that use recursive functions to traverse the given binary tree in
 - a) Pre-order
 - b) In-order
 - c) Post-order

WEEK 10:

10. Write a C program to implement Binary Search Tree with the following operations:
 - a) Insert an element into a binary search tree.
 - b) Delete an element from a binary search tree.
 - c) Search for a key element in a binary search tree.

WEEK 11:

11. Write a C program to implement BFS and DFS for a given graph.

WEEK 12:

12. Write a C program to implement the following Hashing techniques.
 - a) Linear Probing
 - b) Quadratic Probing
 - c) Double Hashing

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

II B.Tech. I-Sem (CSE)

P	C
3	2

(A0593123) DIGITAL LOGIC DESIGN LAB

(Common to CSE & IT)

1. Basic Logic Gates AND,OR,NOT and their applications
2. Universal gates NAND and NOR
3. Study of combinational circuits 1 Half Adder and Full Adder
4. Study of combinational circuits 1 Half Sub tractor and Full sub tractor.
5. Study of Flip flops
 - a) S-R F/F b) J-K F/F c) D-F/F d)T - F/F
6. Design of four bit ring counter using Flip Flop
7. 3 – bit synchronous counter using Flip Flop
8. 4-bit Johnson Ring counter using Flip Flop
9. MOD-5 Synchronous counter using F/F
10. 2-4 decoder
11. 4 to 1 Multiplexer
12. 3 – bit up/down counter using F/F

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

II B.Tech. I-Sem (CSE)

P	C
3	2

(A0294123) PRINCIPLES OF ELECTRICAL AND ELECTRONICS ENGINEERING LAB**OBJECTIVE:**

- It helps to practically verify the basic concepts of electrical & electronic principles
- It helps the student to conceptualize the characteristics of basic electronic devices and their performance

OUTCOMES:

- The student will get clear understanding over the theoretical concepts through experimentation

The following experiments are required to be conducted as compulsory experiments:

1. Verification of KVL & KCL
2. Verification of Superposition theorem.
3. Verification of Thevenin's theorem
4. Verification of Active and reactive power in Resistive and inductive circuits
5. Verification of Maximum power transfer theorem.
6. V-I characteristics of P-N diode
7. Verification of output Wave form of Half Wave Diode Rectifier.
8. Verification of output Wave form of Full Wave Bridge Diode Rectifier.
9. V-I Characteristics of common Emitter configuration
10. Frequency response of CE amplifier

(A0011123) MANAGERIAL ECONOMICS & FINANCIAL ANALYSIS

(Common to all Branches)

OBJECTIVES

- To understand the principles of and techniques of managerial economics.
- To understand the business organizations.
- To understand the financial accounting and analysis.
- To understand maintain a particular product at the lowest cost while meeting the specifications of the customer

OUTCOMES:

- Students will able to analyse the demand in the present market.
- Students will able to how to precise the production cost.
- Students will able to know the price output decisions are made in markets.
- Students will able to maintain the books by using the financial accounting

UNIT I

Introduction to Managerial Economics: Definition, Nature and Scope of Managerial Economics–Demand Analysis: Demand determinants, Law of Demand and its exceptions.

UNIT II

Elasticity of Demand: Definition, Types, Measurement and Significance of Elasticity of Demand. Demand Forecasting, Factors governing demand forecasting, methods of demand forecasting (survey methods, statistical methods, expert opinion method, test marketing, controlled experiments, judgmental approach to demand forecasting)

UNIT III

Business & New Economic Environment: Characteristic features of Business, Features and evaluation of Sole Proprietorship, Partnership, Joint Stock Company, Public Enterprises and their types, Changing Business Environment in Post-liberalization scenario.

UNIT IV

Capital and Capital Budgeting: Capital and its significance, Types of Capital, Estimation of Fixed and Working capital requirements, Methods and sources of raising finance.

Nature and scope of capital budgeting, features of capital budgeting proposals, Methods of Capital Budgeting: Payback Method, Accounting Rate of Return (ARR) and Net Present Value Method (simple problems).

UNIT V

Introduction to Financial Accounting: Double-Entry Book Keeping, Journal, Ledger, Trial Balance- Final Accounts (Trading Account, Profit and Loss Account and Balance Sheet with simple adjustments).

UNIT VI

Financial Analysis through ratios: Computation, Analysis and Interpretation of Liquidity Ratios (Current Ratio and quick ratio), Activity Ratios (Inventory turnover ratio and Debtor Turnover ratio), Capital structure Ratios (Debt- Equity ratio, Interest Coverage ratio), and Profitability ratios (Gross Profit Ratio, Net Profit ratio, Operating Ratio, P/E Ratio and EPS).

TEXT BOOKS:

1. Aryasri: Managerial Economics and Financial Analysis, 2/e, TMH, 2005.
2. Varshney & Maheswari: Managerial Economics, Sultan Chand, 2003.

REFERENCES:

1. Ambrish Gupta, Financial Accounting for Management, Pearson Education, New Delhi.
2. H. Craig Peterson & W. Cris Lewis, Managerial Economics, PHI, 4th Ed.
3. Suma Damodaran, Managerial Economics, Oxford University Press.
4. Lipsey & Chrystel, Economics, Oxford University Press.
5. S. A. Siddiqui & A. S. Siddiqui, Managerial Economics & Financial Analysis, New age International Space Publications.
6. Domnick Salvatore: Managerial Economics In a Global Economy, 4th Edition, Thomson.
7. Narayanaswamy: Financial Accounting—A Managerial Perspective, PHI.
8. Raghunatha Reddy & Narasimhachary: Managerial Economics& Financial Analysis, Scitech.
9. S.N.Maheswari & S.K. Maheswari, Financial Accounting, Vikas.
10. Truet and Truet: Managerial Economics: Analysis, Problems and Cases, Wiley.Dwivedi:Managerial Economics, 6th Ed., Vikas.

Codes/Tables: Present Value Tables need to be permitted into the examinations Hall.

(A0505124) COMPUTER SYSTEM DESIGN**OBJECTIVES:**

- To understand the structure, function, characteristics and performance issues of computer systems.
- To understand the design of the various functional units of digital computers.
- To understand the basic processing unit and how they are connected and how it generates control signals (using hardwired and micro programmed approaches).
- To understand the architecture of 8086
- To understand the different types of memory and how they are related.
- To learn basics of Parallel Computing and Pipelining.

OUTCOMES:

- Students will learn about computer performance, computer design, and trade-offs between cost and performance as well as between hardware and software
- Students will formulate and solve problems, understand the performance requirements of systems
- Students will learn to communicate effectively and learn to think creatively and critically, both independently and with others.
- Students will learn about all the detailed design issues and circuits of each unit.

UNIT I

BASIC STRUCTURE OF COMPUTERS: Computer Types, Functional unit, Basic operational concepts, Bus structures, Software, Performance, multiprocessors and multi computers.

DATA REPRESENTATION: Fixed Point Representation, Floating Point Representation. Error Detection codes.

UNIT II

REGISTER TRANSFER LANGUAGE AND MICROOPERATIONS: Register Transfer language, Register Transfer, Bus and memory transfers, Arithmetic Micro operations, logic micro operations, shift micro operations

UNIT III

BASIC COMPUTER ORGANIZATION AND DESIGN: Instruction codes, Computer Registers, Computer instructions, Instruction cycle, Memory- reference instructions, Input – Output and Interrupt.

COMPUTER ARITHMETIC: Fixed point operations - Addition and subtraction, multiplication, Division Algorithms

UNIT IV

8086: Register organization, Architecture, Pin diagram, Addressing modes ,importance of segment registers, Evaluation of Physical address **Assembly Language Programming instructions: MOV, PUSH, POP, XCHG, IN, OUT, LEA, CMP, AAA, AAS, AAM, AAD, SHIFT(No programs)**

UNIT V

THE MEMORY SYSTEM: Basic concepts, semiconductor RAM memories, Read-only memories, Cache memories, performance considerations.

UNIT-VI

PIPELINE AND VECTOR PROCESSING: Parallel processing, Arithmetic pipeline, Instruction Pipeline, RISC Pipeline, Vector processing, Array Processors.

TEXT BOOKS

1. Computer Systems Architecture – M. Moris Mano, IIIrd Edition, Pearson/PHI
2. Computer Organization – Carl Hamacher, Zvonks Vranesic, SafeaZaky, Vth Edition, McGraw Hill.

REFERENCES

1. Advanced Microprocessors and peripherals by A K RAY

(A0506124) OBJECT ORIENTED PROGRAMMING

(Common to CSE, ECE & EEE)

OBJECTIVES:

After taking this course, the student should be able to:

- Describe the Windows event-driven programming model
- Build simple JAVA applications according to the model
- Write fluent JAVA code for creating classes
- Use JAVA variables, data, expressions and arrays
- Design and create forms, menus and controls
- Write clear, elementary Java programs (applets and applications)
- Use a Java-enabled browser and/or the applet viewer to execute Java applets
- Use the Java interpreter to run Java applications
- Design and construct effective graphic user interfaces for application software.
- Use Java Beans, RMI to build complex business applications

OUTCOMES:

- Understand the syntax and concepts of JAVA
- Write JAVA programs for processing data
- Write JAVA programs to interface with windows.
- Write JAVA programs that use data from flat files and databases.
- Develop programs with GUI features such as dialog boxes, menus etc.
- Write JAVA programs that form the GUI front-end for database applications.
- Write applications using distributed objects.
- A passing student shall demonstrate knowledge of GUI-based event-driven programming in a working.
- Program assignment utilizing Java GUI components, event listeners and event-handlers.

UNIT I

Introduction To Java – Introduction to OOP, OOP Concepts, History of Java, Java buzzwords, How Java differs from C , Structure of Java Program, data types, variables, constants, type conversion and casting, enumerated types, scope and life time of variables, operators, expressions , control flow- conditional statements, break and continue, simple java program, arrays, parameter passing, static fields and methods, access control, this, overloading methods and constructors, recursion, garbage collection.

UNIT II

Inheritance –Inheritance concept, Super and Sub classes, Member access rules, types of Inheritance, super uses, final classes and methods, casting, polymorphism- dynamic binding, method overriding, abstract classes and methods, the Object class and its methods.

UNIT III

Interfaces – Interfaces vs. Abstract classes, defining an interface, implementing interfaces, accessing implementations through interface references, extending interface.

Packages- Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages.

UNIT IV

Files – streams, text Input/output, binary input/output, random access file operations, File management using File class, Using java.io.

Strings: Strings, string functions.

UNIT V

Exception handling – benefits of exception handling, exception hierarchy, checked exceptions and unchecked exceptions, usage of try, catch, throw, throws and finally, rethrowing exceptions, built in exceptions, creating own exceptions.

Multithreading - Differences between multiple processes and multiple threads, thread states, creating threads, interrupting threads, thread priorities, synchronizing threads, interthread communication, thread groups, daemon threads, thread deadlock.

UNIT VI

Event Handling - Events, Event sources, Event classes, Event Listeners, Relationship between Event sources and Listeners, Delegation event model, handling mouse and keyboard events, Adapter classes.

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**TEXT BOOKS**

1. Java; the complete reference, 7th edition, Herbert Schildt, TMH.
2. Understanding OOP with Java, updated edition, T. Budd, Pearson Education.

REFERENCES

1. An Introduction to programming and OO design using Java, J.Nino and F.A. Hosch, John Wiley & Sons.
2. An Introduction to OOP, second edition, T. Budd, Pearson Education.
3. Introduction to Java programming 6th edition, Y. Daniel Liang, Pearson Education.
4. An introduction to Java programming and object oriented application development, R.A. Johnson-Thomson.
5. Core Java 2, Vol 1, Fundamentals, Cay.S.Horstmann and Gary Cornell, seventh Edition, Pearson Education.
6. Core Java 2, Vol 2, Advanced Features, Cay.S.Horstmann and Gary Cornell, Seventh Edition, Pearson Education.

(A0507124) FORMAL LANGUAGES AND AUTOMATA THEORY**OBJECTIVES:**

The purpose of this course is to acquaint the student with an overview of the theoretical foundations of computer science from the perspective of formal languages.

- Classify machines by their power to recognize languages.
- Employ finite state machines to solve problems in computing.
- Explain deterministic and non-deterministic machines.
- Comprehend the hierarchy of problems arising in the computer sciences.

OUTCOMES:

- To introduce the computer science students to the theoretical foundations of computer science.
- To study abstract models of information processing machines and limits of digital computation.
- To provide theoretical preparation for the study of programming languages and compilers.
- To develop the skills of formal and abstract reasoning as needed; for example, when designing, analysing, and / or verifying complex software/hardware systems.

UNIT I:

Automata: The Methods and Madness-Why study Automata Theory: introduction to FA, Structural Representations, Automata and complexity. **The central concepts of Automata:** Alphabet, Strings, Language, Operations, Finite state machine. **Deterministic finite automaton:** definition, How DFA processes Strings, Simpler notations for DFA's, Extending the transition Function to Strings, The Language of a DFA. **Non deterministic finite automaton:** An informal view of NFA, Definition, The extended Transition Function, The language of a NFA, An Application.

Finite Automata with ϵ transitions: use of ϵ transitions, Notation, Epsilon closures, Eliminating ϵ Transitions. NFA to DFA conversion, minimization of FSM, **Finite Automata with output:** Moore and Melay machines.

UNIT II:

Regular Languages: Regular sets, regular expressions, identity rules for regular expressions, Arden's Theorem.

Finite Automata and regular expressions: Transition systems and regular expressions, Transition system containing ϵ moves, Algebraic method using Arden's theorem, Conversion of Finite Automata equivalent to Regular expressions, Equivalence of two finite automata. Pumping lemma of regular sets, Applications of pumping lemma, closure properties of regular sets (proofs not required). **Regular sets and regular grammars:** construction of Regular grammar generating for a given DFA.

UNIT III:

Context free grammar and Languages- CFG: Definition of CFG, Right most and leftmost derivations, The language of a grammar, Sentential Forms. **Parse Trees:** Constructing Parse Trees, Application of CFG. **Ambiguity in grammars and Languages:** Ambiguous grammars.

Properties of CFL-Normal Forms for CFG: Eliminating Useless Symbols, Computing the generating and reachable symbols, Eliminating ϵ - productions, Eliminating unit productions, Chomsky normal form, Greibach normal form, Pumping Lemma for Context Free Languages. Closure properties of CFL (proofs omitted).

UNIT IV:

Push Down Automata- PDA: definition, model, graphical notation for PDA's, Instantaneous Descriptions of a PDA. **The Language of a PDA:** acceptance by final state and acceptance by empty state and its equivalence. Equivalence of CFL and PDA, inter conversion. (Proofs not required). Introduction to DCFL and DPDA.,

UNIT V

Turing Machine: Chomsky hierarchy of languages **Turing assumptions,** Turing Machine as computational machine, Techniques for Turing machine construction, Types of Turing machines.

UNIT VI

Church's thesis, Turing machines as enumerators, Universal Turing Machine, Counter machine Recursive and recursively enumerable languages, linear bounded automata and context sensitive language.

TEXT BOOKS:

1. "Introduction to Automata Theory Languages and Computation". John E. Hopcroft, Rajeev Motwani and Jeffery D. Ullman. Pearson Education – third edition
2. Formal Language and Automata Theory by KVN Sunitha ,Tata McGraw Hill Education

REFERENCES:

1. "Theory of Computer Science (Automata languages and computation)" K. L. P. Mishra and N. Chandra Shekaran, 2nd edition, PHI.

(A0508124) DATABASE MANAGEMENT SYSTEMS**OBJECTIVES:**

- Advantages applications of DBMS and Database system structure.
- Schema design: ER model and conceptual design.
- Relational model and SQL basics.
- Relational algebra and Query optimization.
- Storage and efficient retrieval of data: various indexing techniques.
- Schema refinement: normalization and redundancy removal and functional dependant.
- Transaction management: locking protocols, serializability concepts etc.
- Concurrency control and crash recovery: various mechanisms, ARIES algorithm and deadlock concepts.

OUTCOMES:

- Students will learn about the need for DBMS, the largeness of the data and why it gives rise to steam oriented processing and strategies and are at higher level than general purpose programming language such as JAVA.
- Students will learn about storage and efficient retrieval of large Information via algebraic query optimization and the use of indexing.
- Students will also learn basics of SQL and about primary key concepts and foreign key concepts. They will also learn about data manipulation (insertions deletions & updation) and triggers.
- Students will learn about functional dependency and the need for schema refinement (normalization) to remove redundancy of data.
- Students will also learn about transaction management concurrency Control and crash recovery.

UNIT I

Database System Applications, database System VS file System – View of Data – Data Abstraction –Instances and Schemas – data Models – the ER Model – Relational Model – Database Languages – DDL – DML – Database Access for applications Programs – Database Users and Administrator – Transaction Management – Database System Structure – Storage Manager – the Query Processor- Data base design and ER diagrams – Beyond ER Design- Entities, Attributes and Entity sets – Relationships and Relationship sets – Additional features of ER Model – Conceptual Design with the ER Model.

UNIT II

Introduction to the Relational Model – Integrity Constraint Over relations – Enforcing Integrity constraints – Querying relational data – Logical database Design – Introduction to Views – Destroying /altering Tables and Views. Relational Algebra – Selection and projection set operations – renaming – Joins – Division – Examples of Algebra queries – Relational calculus – Tuple relational Calculus – Domain relational calculus.

UNIT III

The Form of a Basic SQL Query – Examples of Basic SQL Queries – Introduction to Nested Queries – Correlated Nested Queries, Set – Comparison Operators – Aggregate Operators – NULL values – Comparison using Null values – Logical connectivity's – AND, OR and NOT – Impact on SQL Constructs – Outer Joins – Disallowing NULL values – Complex Integrity Constraints in SQL, Triggers and Active Data bases.

UNIT IV

Schema refinement – Problems Caused by redundancy – Decompositions – Problems related to decomposition – Functional dependencies-reasoning about FDS – FIRST, SECOND, THIRD Normal forms – BCNF – Lossless join Decomposition – Dependency preserving Decomposition – Schema refinement in Data base Design – Multi valued Dependencies – FORTH Normal Form.

UNIT V

Overview Of Transaction Management: The ACID Properties, Transactions and Schedules, Concurrent Execution of transactions-Lock Based Concurrency Control, Performance of Locking, Transaction Support in SQL, Introduction to crash recovery, Log – Based Recovery – Recovery with Concurrent Transactions– Failure

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

with loss of non-volatile storage-Remote Backup systems. Concurrency Control: 2PL, serializability and recoverability, Introduction Lock Management, Lock Conversions, Dealing with Deadlocks, Concurrency control without locking.

UNIT VI

Data on External Storage – File Organizations and Indexing – Cluster Indexes, Primary and Secondary Indexes – Index data Structures – Hash Based Indexing – Tree base Indexing – Comparison of File Organizations – The Memory Hierarchy, RAID, Disk Space Management, Buffer Manager, Files of Records, Page Formats, record Formats.

TEXT BOOKS:

1. Data base Management Systems, Raghu Ramakrishna, Johannes Gehrke, TATA Mc Graw Hill 3rd Edition
2. Data base System Concepts, Silberschatz, Korth, McGraw hill, V edition.

REFERENCES:

1. Data base Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
2. Fundamentals of Database Systems, Elmasri Navathe Pearson Education.
3. Introduction to Database Systems, C.J.Date Pearson Education

(A0509124) OPERATING SYSTEMS

(Common to CSE & ECE)

OBJECTIVES:

- This course deals with functions, structures and history of operating systems.
- To understand the design issues associated with operating systems.
- To understand various process management concepts including scheduling, synchronization, deadlocks.
- To be familiar with multithreading and the concepts of memory management including virtual memory.
- To understand the issues related to file system interface and implementation, disk management with protection and security mechanisms.
- Some example operating systems (Unix, Windows, Solaris etc)

OUTCOMES:

- At the end of the course the students knows the need and requirement of an interface between Man and Machine.
- To enable them to identify the difference between the system software and the application software and their design requirements.
- Students will be able to relate the features of operating systems and the fundamental theory associated with process, memory and file managements components of different operating systems.
- Students will learn about and understand theoretical concepts and programming constructs used for the operation of modern operating systems.
- Students will gain practical experience with software tools available in modern operating systems such as semaphores, system calls, sockets and threads

UNIT I

Introduction - what operating systems do, process management, memory management, protection and security, distributed systems, special purpose systems

System structure - operating system services, systems calls, types of system calls, system programs, operating system structure, operating systems generation, system boot.

UNIT II

Process concepts – overview, process scheduling, operations on process, inter-process communication.

Multithread Programming – overview, multithreading models, thread libraries.

Process scheduling – basic concepts, scheduling criteria, process scheduling algorithms, algorithm evaluation.

UNIT III

Concurrency - Process synchronization, the critical-section problem, Peterson’s Solution, synchronization Hardware, semaphores, classic problems of synchronization, monitors

Principles of deadlock: system model, deadlock characterization, deadlock prevention, detection and avoidance, recovery form deadlock.

UNIT IV

Memory Management Strategies - Swapping, contiguous memory allocation, paging, structure of the page table, segmentation

Virtual memory management – background, demand paging, copy-on-write, page-replacement algorithms, Thrashing.

UNIT V

File system – file concept, Access Methods, Directory structure, protection.

File System implementation - File system structure, file system implementation, directory implementation, allocation methods, free-space management, Recovery.

UNIT VI

Secondary-storage structure- overview of Mass-storage structure, Disk structure, disk attachment, disk scheduling, swap-space management, RAID structure, stable-storage implementation

Tertiary storage devices-removable disks, tapes, future technology, performance issues.

TEXT BOOKS:

1. Operating System Concepts, Abraham Silberchatz, Peter B. Galvin, Greg Gagne, Eighth edition, John Wiley.

REFERENCES:

1. Operating Systems: Internals and Design Principles, Stallings, Sixth Edition–2009, Pearson Education.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

II B.Tech. II-Sem (CSE)

T	C
3	2

(A0007123)APTITUDE ARITHMETIC REASONING AND COMPREHENSION

(Common to All Branches)

(Skill Development Course)**OBJECTIVES:**

- To make the students ready to the recruitment drives.
- To raise the confidence of the students to face the written test of any Company.
- To train the students regarding employability skills.

OUTCOMES:

- Students becomes well trained for recruitment drives.
- Student become well trained to face the written test of any company.
- Students become well trained in employability skills

UNIT I

Numbers, Number Systems Simple Equations, Ratio, Proportion, Variation Quadratic Equations, Progressions Percentages.

UNIT II

Profit, Loss, Partnerships Averages, Mixtures & Allegations, Simple Interest, Compound Interest, Time and Work-Pipes, indices, surds, inequalities ,Cisterns Time and Distance Geometry and Menstruation.

UNIT III

Permutations & Combinations and Probability Data Interpretation & Data Sufficiency.

UNIT IV

Number & Letter Series, Analogies, Coding Decoding, Odd Man Out Blood Relations.

UNIT V

Direction Sense, Symbols and Notations Deductions & Connectives Clocks, Calendars Analytical

UNIT VI

Reasoning (Distribution+ Binary Logic + Puzzles) Cubes, Venn Diagrams Analytical Puzzles (Linear + Circular +Selections + Sequencing + Routes & Networks + Comparisons) and Non Verbal Reasoning

REFERENCES:

1. R.S.Agarwal “ Quantitative Techniques” S.Chand Series
2. Shankuntala Devi “ Techniques of Reasoning” S.Chand Series

(A0594124) OPERATING SYSTEMS LAB**OBJECTIVES:**

A student will be able to:

- Prepare students for easy transfer from academia into practical life.
- Get an Insight into the Computer Technologies.
- Obtain Basic Knowledge of Operating Systems.

OUTCOMES:

The main learning outcomes are:

- Master functions, structures and history of operating systems
- Master understanding of design issues associated with operating systems
- Master various process management concepts including scheduling, synchronization, deadlocks
- Be familiar with multithreading
- Master concepts of memory management including virtual memory
- Master system resources sharing among the users
- Master issues related to file system interface and implementation, disk management
- Be familiar with protection and security mechanisms
- Be familiar with various types of operating systems including Unix

1. Simulate the following CPU scheduling algorithms
 - a) FCFS b) SJF c) Priority d) Round Robin
2. Simulate Bankers Algorithm for Dead Lock Avoidance
3. Simulate Bankers Algorithm for Dead Lock Detection
4. Simulate the page replacement algorithms
 - a) FIFO b) LRU c) Optimal
5. Simulate the dynamic storage allocation strategies
 - a) First fit b) Best fit
6. Simulate MVT architecture
7. Simulate MFT architecture
8. Simulate sequential file allocation strategy
9. Simulate following File Organization Techniques
 - a) Single level directory b) Two level directory

(A0595124) OBJECT ORIENTED PROGRAMMING LAB

(Common to CSE and ECE)

OBJECTIVES:

- To make the student operating systems.
- Learn a object oriented way of solving problems.
- To teach the student to write programs in Java to solve the problems

OUTCOMES:**After Completion of the Lab Course student should be able:**

- To make the student learn a object oriented way of solving problems.
- To teach the student to write programs in Java to solve the problems

Recommended Systems/Software Requirements:

- Intel based desktop PC with minimum of 166 MHZ or faster processor with at least 64 MB RAM and 100 MB free disk space.
 - JDK Kit. Recommended
- 1) a) Write a Java program that prints all real solutions to the quadratic equation $ax^2 + bx + c = 0$. Read in a, b, c and use the quadratic formula. If the discriminant $b^2 - 4ac$ is negative, display a message stating that there are no real solutions.
 - b) The Fibonacci sequence is defined by the following rule:
The first two values in the sequence are 1 and 1. Every subsequent value is the sum of the two values preceding it. Write a Java program that uses both recursive and non recursive functions to print the nth value in the Fibonacci sequence.
 - 2) a) Write a Java program that prompts the user for an integer and then prints out all prime numbers up to that integer.
 - b) Write a Java program to multiply two given matrices.
 - 3) Write a Java Program that reads a line of integers, and then displays each integer, and the sum of all the integers (Use String Tokenizer class of java. until)
 - 4) Write a Java program to find both the largest and smallest number in a list of integers.
 - 5) Write a Java program to illustrate method overloading.
 - 6) Write a Java program that implements the Sieve of Eratosthenes to find prime numbers.
 - 7) Write a Java program to sort a list of names in ascending order.
 - 8) Write a Java program to implement the matrix ADT using a class. The operations supported by this ADT are:
 - a) Reading a matrix.
 - b) Printing a matrix.
 - c) Addition of matrices.
 - d) Subtraction of matrices.

Write a Java Program to solve Tower's of Hanoi problem .
 - 9) Write a Java Program that uses a recursive function to compute ncr. (Note: n and r values are given)
 - 10) Write a Java program to perform the following operations:
 - a) Concatenation of two strings.
 - b) Comparison of two strings.
 - 11) Implement the complex number ADT in Java using a class. The complex ADT is used to represent complex numbers of the form $c=a+ib$, where a and b are real numbers. The operations supported by this ADT are:
 - a) Reading a complex number.
 - b) Writing a complex number.
 - c) Addition of Complex numbers.
 - d) Subtraction of complex numbers.
 - e) Multiplication of complex numbers.
 - f) Division of complex numbers.

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

- 12) Write a Java program that makes frequency count of letters in a given text.
- 13) Write a Java program that uses functions to perform the following operations :
- Inserting a sub-string in to the given main string from a given position.
 - Deleting n characters from a given position in a given string.
- 14) a) Write a Java program that checks whether a given string is a palindrome or not. Ex:
MADAM is a palindrome.
- b) Write a Java program to make frequency count of words in a given text.
- 15) a) Write a Java program that reads a file name from the user, then displays information about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes.
- b) Write a Java program that reads a file and displays the file on the screen, with a line number before each line.
- c) Write a Java program that displays the number of characters, lines and words in a text file.
- d) Write a Java program to change a specific character in a file.
- Note:** Filename, number of the byte in the file to be changed and the new character are specified on the command line.
- 16) Write a Java program that:
- Implements stack ADT.
 - Converts infix expression into Postfix form
 - Evaluates the postfix expression.
- 17) a) Write a Java program that creates three threads. First thread displays “Good Morning” every one second, the second thread displays “Hello” every two seconds and the third thread displays “Welcome” every three seconds.
- b) Write a Java program that correctly implements producer consumer problem using the concept of inter thread communication.

II B.Tech. II-Sem (CSE)

P C
3 2**(A0596124) DATABASE MANAGEMENT SYSTEMS LAB****OBJECTIVES**

- Student will be able to:
- Create and delete database schemas and execute SQL queries
- Inserting data, Altering and dropping the tables.
- Various types of data conversions using the functions.
- Make Use of PL/SQL Language Components.
- Make Use of PL/SQL Variables.
- Handle PL/SQL Reserved Words.
- Make Use of Identifiers in PL/SQL
- Make Use of Anchored Data type

OUTCOMES

Upon completion of the lab, the student should be able to:

- Map the model into a relational database system.
- Implement the given schema on a relational DBMS.
- Design, develop, and maintain Oracle Database Objects.
- Use a database language for manipulating and querying data.
- Develop advanced packages, stored procedures, and triggers and functions using PL/SQL

Recommended Systems/Software Requirements:

- Intel based desktop PC
- Mysql /Oracle latest version Recommended.

- 1) Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.
- 2) Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSET, Constraints.
Example:- Select the roll number and name of the student who secured fourth rank in the class.
- 3) Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.
- 4) Nested Queries and correlated nested queries
- 5) Table alterations
- 6) Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next_day, add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date)
- 7) Creating procedures
- 8) Creating functions and packages
- 9) Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers.
- 10) Introduction to ORACLE reports

TEXT BOOKS :

1. ORACLE PL/SQL by example. Benjamin Rosenzweig, Elena Silvestrova, Pearson Education 3rd Edition
2. ORACLE DATA BASE LOG PL/SQL Programming SCOTT URMAN, Tata Mc-Graw Hill.
3. SQL & PL/SQL for Oracle 10g, Black Book, Dr.P.S. Deshpande.

(A0510125) ADVANCED JAVA PROGRAMMING**OBJECTIVES:**

- This course demonstrates an in-depth understanding of the tools and Web technologies necessary for business application design and development. The course covers client side scripting like HTML, JavaScript and server side scripting like servlets, JSPs. And also XML and web servers and database interfacing.

OUTCOMES:

- The main learning outcomes are:
- Development of a business application.
- Implementation of given client side and server side technologies.
- Design and develop static and dynamic web pages.
- Validate web page data with database data.

UNIT I

Applet Basics: Applet class, Applet Architecture, Applet skeleton, Simple Applet display methods, Requesting Repainting, The HTML APPLETTAG, passing parameters to Applets, Applet Context & show Document.

UNIT II

Networking in Java – Introduction, Manipulating URLs, Ex. Client/Server Interaction with Stream Socket Connections, Connectionless Client/Server Interaction with Datagrams, Using java.net.

UNIT III

GUI Programming with Java - The AWT class hierarchy, Introduction to Swing, Swing vs. AWT, MVC architecture, Hierarchy for Swing components, Containers – Top-level containers – JFrame, JApplet, JWindow, JDialog, JPanel, A simple swing application, swing components- JButton, JToggleButton, JCheckBox, JRadioButton, JLabel, JTextField, JTextArea, JList, JComboBox, JMenu, capabilities –color control, Font control, Drawing lines, rectangles and ovals, Drawing arcs, Layout management - Layout manager types – border, grid, flow, box.

UNIT IV

Java Beans: Introduction to Java Beans, Advantages of Java Beans, Bean Development Kit, JAR Files, JDK Introspection, Using Bound properties, Bean Info Interface, Constrained properties, Persistence, Developing a Simple Bean.

UNIT V

Web Servers and Servlets: Installing the Java Software Development Kit, Tomcat Server & Testing Tomcat Introduction to Servlets: Lifecycle of a Servlet, A Simple servlet, The Servlet API, The javax. servlet Package, Reading Servlet Parameters.

UNIT VI

Introduction to JSP: The Problem with Servlet. The Anatomy of a JSP Page, JSP Processing. JSP Application Design with MVC Setting Up and JSP Environment.

TEXT BOOKS:

1. Java; the complete reference, 7th edition, Herbert Schildt, TMH.
2. Java Server Pages –Hans Bergsten, SPD O'Reilly

REFERENCE BOOKS:

1. Introduction to Java programming 6th edition, Y. Daniel Liang, Pearson Education.
2. An introduction to Java programming and object oriented application development, R.A. Johnson-Thomson.
3. Core SERVLETS AND JAVASERVER PAGES VOLUME1: Core Technologies By Marty Hall and Larry Brown Pearson.
4. SDFS Murach's beginning JAVA JDK 5, Murach, SPD.

(A0511125) COMPUTER NETWORKS

(Common to CSE & ECE)

OBJECTIVES:

- An understanding of the overriding principles of computer networking, including protocol design, protocol layering, algorithm design, and performance evaluation.
- An understanding of computer networking theory, including principles embodied in the protocols designed for the application layer, transport layer, network layer, and link layer of a networking stack.
- An understanding of specific implemented protocols covering the application layer, transport layer, network layer, and link layer of the Internet (TCP/IP) stack.
- An understanding of security issues.

OUTCOMES:

- Students will learn to list and classify network services, protocols and architectures, explain why they are layered.
- Student will learn to explain key Internet applications and their protocols.
- Students will learn to explain security issues in computer networks.
- To master the terminology and concepts of the OSI reference model and the TCP-IP reference model.
- To master the concepts of protocols, network interfaces, and Design/performance issues in local area networks and wide area networks.
- To be familiar with wireless networking concepts.
- To be familiar with contemporary issues in networking technologies.
- To be familiar with network tools and network programming.

UNIT I:

Introduction: Network Hardware, Network Software, References Models. **The Physical Layer:** Guided Transmission Media, Communication Satellites, The public Switched Telephone Network: Trunks and Multiplexing, Switching

UNIT II:

The Data Link Layer: Data link Layer Design Issues, Elementary Data Link Protocols, Sliding Window Protocols. Error Detection and Correction.

UNIT III:

The Medium Access Control Sub layer: Multiple Access protocols, Ethernet- Ethernet Cabling, Manchester Encoding, The Ethernet MAC Sub layer Protocol. The Binary Exponential Back off Algorithm, Ethernet Performance, Switched Ethernet, Fast Ethernet. Wireless LANs- The 802.11 Protocol Stack, The 802.11 Physical Layer, The 802.11 MAC Sub Layer Protocol, The 802.11 Frame Structure.

UNIT IV:

The Network Layer: Network Layer Design Issues, Routing Algorithms (Shortest path, Flooding, Distance Vector, Link state and Hierarchical routing, Broad cast routing, Multicast routing), Congestion Control Algorithms, Internetworking.

UNIT V:

The Transport Layer: The Transport Service, Elements of Transport Protocols, The Internet Transport Protocols: UDP, The Internet Transport Protocols: TCP.

UNIT VI:

The Application Layer: DNS-The Domain Name System, Electronic Mail, The World Wide Web: Architectural overview. Definitions of encryption, Decryption, Advantages of network security.

TEXT BOOKS:

1. Computer Networks- Andrew S. Tanenbaum, Fourth Edition, Pearson Education.

REFERENCES:

1. Data Communications and Networking, Behrouz A. Forouzan, Fourth Edition, Tata McGraw Hill.

(A0512125) PRINCIPLES OF PROGRAMMING LANGUAGES**OBJECTIVES:**

- The main objective is that students will have a deep, working knowledge of the functional paradigm and the key ideas used in modern programming languages. It provides an exposure to core concepts and principles in contemporary programming languages

OUTCOMES:

- After the completion of this course, students should be able to:
- Write and modify programs using a mostly-functional style.
- Write and modify programs that make effective use of data abstraction.
- Modify interpreters to change or enhance their behavior so as to implement various features of programming languages.
- Write programs using such features, and explain, using appropriate terminology.

UNIT I

Preliminary Concepts: Reasons for studying concepts of programming languages, Programming domains, Language Evaluation Criteria Influences on language design, Language categories, Language Design Trade-Offs, Implementation Methods, Programming Environments Syntax and Semantics: general Problem of describing Syntax and Semantics, Formal Methods of Describing Syntax-BNF,EBNF.

UNIT II

Data types: Introduction, primitive, character, user defined, array, associative, record, union, design and implementation uses related to these types. Names, Variable, type checking, strong typing, type compatibility.

UNIT III

Expressions and Statements: Arithmetic relational and Boolean expressions, Short circuit evaluation Assignment Statements, Control Structures: Statement Level, Compound Statements, Selection, Iteration, guarded commands.

UNIT IV

Subprograms and Blocks: Fundamentals of sub-programs, Scope and lifetime of variable, Design issues of subprograms and operations, parameter passing methods, generic sub-programs, user defined overloaded operators, co routines.

UNIT V

Abstract Data types: Abstractions and encapsulation, introductions to data abstraction, language examples, object oriented programming in small talk, C++, Java, C#. Exception handling: Exceptions, exception Propagation, Exception handler in Java, Ada.

UNIT VI

Logic Programming Language: Introduction and overview of logic programming, basic elements of prolog, application of logic programming. Functional Programming Languages: Introduction, fundamentals of FPL, The first FPL: LISP, ML, Haskell, application of Functional Programming Languages and comparison of functional and imperative Languages.

TEXT BOOKS:

1. Concepts of Programming Languages Robert W. Sebesta, Eighth Edition, Pearson Education, 2008.
2. Programming Language Design Concepts, D. A. Watt, Wiley Dreamtech, rp-2007.

REFERENCES:

1. Programming Languages, Second Edition, A.B. Tucker, R.E. Noonan, TMH.
2. Programming Languages, K. C.Louden, Second Edition, Thomson, 2003.
3. LISP, Patric Henry Winston and Paul Horn, Pearson Education.
4. Programming in Prolog, W.F. Clocksin and C.S.Mellish, Fifth Edition, Springer.
5. Programming Python, M.Lutz, Third Edition, O'reilly, SPD, rp-2007.

(A0513125) COMPILER DESIGN**OBJECTIVES:**

- Thorough understanding of the overall architecture of a modern compiler.
- Being familiar with both top-down and bottom-up parsing paradigms.
- Fluent with syntax-directed translation scheme and different compiler-compilers.
- Knowledgeable with assembly language and code-block based code generation scheme.
- Knowing the inner details of compilers, libraries, operating systems/platforms, and how they interact with each other to form modern computing environments.

OUTCOMES:

- After the completion of this course, the students will be able to: describe the theory and practice of compilation.
- Generally applying the top down and bottom approaches of parsing, and the lexical analysis.
- Student will be able to generate code generation and optimization phases of compilation, and design a compiler for a concise programming language.

UNIT-I

Introduction: Language processors, structure of a compiler: Lexical analysis, syntax analysis, semantic analysis, intermediate code generation, code optimization, code generation, symbol table management, the grouping of phases into passes. Definition of Assembler, pre-processor, compiler, Natural language processor and interpreter.

UNIT-II

Syntax analysis- introduction: the role of the parser. CFG: definition, notation, derivation, parse tree, ambiguity. Writing a grammar: Eliminating ambiguity, elimination of left recursion, left factoring
Top down Parsing: recursive descent Parsing, first and follow, LL (1) grammar.

UNIT-III

Bottom up Parsing: reductions, handle pruning, shift reduce parsing, and conflict during shift reduce parsing. Introduction to LR parsing: SLR, Why LR parsers, Items and LR (0) automaton, LR Parsing Algorithm, constructing SLR parsing tables. More powerful LR Parsers: CLR (1) Items, CLR (1) Parsing tables.

UNIT-IV

Intermediate code generation-variants of syntax trees, DAG for expressions. Three address code: addresses and instructions, quadruples, triples, indirect triples. Types and declarations: type expressions, type equivalence. Type checking: rules for type checking, type conversions.

UNIT-V

Symbol Tables: Symbol table format, organization for block structures languages, hashing, tree structures representation of space information. Block structures and non block structure storage allocation: static, Runtime stack and heap storage allocation, storage allocation for arrays, strings and records. Storage organization: static versus dynamic storage allocation, Activation trees, Activation Records, calling sequences,

UNIT-VI

Code optimization: Consideration for Optimization, Scope of Optimization, local Optimization, loop Optimization, global Optimization, machine dependent code Optimization.
Object code forms; register allocation and assignment, generic code generation algorithms, DAG for register allocation. Data flow analysis.

TEXT BOOKS:

1. Compilers principles, techniques, &tools- A.V.Aho. J.D.Ullman; pearson Education. Second edition.

REFERENCES:

1. Compiler design: Theory, Tools and examples by seth D.Bergmann.

(A0514125) SOFTWARE ENGINEERING**OBJECTIVES:**

- To understand Software development as a process.
- Various software process models and system models.
- Various software designs: Architectural , object oriented , user interface etc.
- Software testing methodologies overview: various testing techniques including white box testing black box testing regression testing etc.
- Software quality: metrics, risk management quality assurance etc.

OUTCOMES:

- Students will learn to work as a team and to focus on getting working project done on time with each student being held accountable for their part of the project.
- Student will learn about risk management and quick prototyping de-risk project management.
- Students will learn about and go through the software life cycle with emphasis on different process requirements design and implementation phases.
- Students will learn about software process models and hoe to choose an Appropriate model for their project will learn about risk management and quick prototyping to de-risk projects.
- Students will gain confidence and conceptualized.

UNIT I

Introduction to Software Engineering: The evolving role of software, Changing Nature of Software, Software myths. **A Generic view of process:** Software engineering- A layered technology, a process framework, The Capability Maturity Model Integration (CMMI), Process patterns, process assessment, personal and team process models.

UNIT II

Process models: The waterfall model, Incremental process models, Evolutionary process models, The Unified process. **Software Requirements:** Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document.

UNIT III

Requirements engineering process: Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management. **System models:** Context Models, Behavioural models, Data models, Object models.

UNIT IV

Design Engineering: Design process and Design quality, Design concepts, the design model.

Creating an architectural design: Software architecture, Data design, Architectural styles and patterns.

UNIT V

Performing User interface design: Golden rules, User interface analysis and design, interface analysis, interface design steps, Design evaluation. **Testing Strategies:** A strategic approach to software testing, test strategies for conventional software, Black-Box and White-Box testing, Validation testing, System testing, the art of Debugging.

UNIT VI

Metrics for Process and Products: Product metrics: Software Quality, Metrics for Analysis Model, Metrics for Design Model, Metrics for source code, Metrics for testing, Metrics for maintenance. Software Quality, Software Measurement, Metrics for software quality. **Quality Management :** Quality concepts, Software quality assurance, Software Reviews, Formal technical reviews, Statistical Software quality Assurance, The ISO 9000 quality standards.

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**TEXT BOOKS:**

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition. McGrawHill International Edition.
2. Software Engineering- Sommerville, 7th edition, Pearson education.

REFERENCES:

1. Software Engineering- K.K. Agarwal & Yogesh Singh, New Age International Publishers
2. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiely.
3. Systems Analysis and Design- Shely Cashman Rosenblatt, Thomson Publications.
4. Software Engineering principles and practice- Waman S Jawadekar, The McGraw-Hill Companies.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

III B.Tech. I-Sem (CSE)

T	C
3+1*	3

(A0515125) C# AND .NET FRAMEWORK

(Common to CSE & IT)

AIM

- To cover the fundamental concepts of the C# language and the .NET framework.

OBJECTIVE:

- The student will gain knowledge in the concepts of the .NET framework as a whole and the technologies that constitute the Framework.
- The student will gain programming skills in C# both in basic and advanced levels.
- By building sample applications, the student will get experience and be ready for large-scale projects.

UNIT I

INTRODUCTION to C# : Introducing C#, Understanding .NET, Overview of C#, Literals, Variables, Data Types, Operators, Expressions, Branching, Looping, Methods, Arrays, Strings, Structures, Enumerations.

UNIT II

OBJECT ORIENTED ASPECTS OF C#: Classes, Objects, Inheritance, Polymorphism, Interfaces, Operator Overloading, Delegates, Errors and Exceptions.

UNIT III

WINDOWS APPLICATIONS: Drawbacks of Console Applications, Container Controls, Non Container Controls, Developing Windows Application from Notepad and Visual Studio, Events, Types of Events – Mouse, Focus, Drag, Key and Other Related Events, Building Windows Applications.

ADO.NET: Problems with File Handling, Data Source Communication, Drivers and Providers, Introduction of ADO.NET, ADO.NET Namespaces, ADO.NET Objects, Accessing Data with ADO.NET.

UNIT IV

BUILDING ASP.NET WEB PAGES: HTML form Development, Client side Scripting, Server side Scripting, Web applications and Web servers, HTTP, Advantages Using ASP.NET, ASP.NET Application, ASP.NET Namespaces, ASP.NET Web Page Structure, Creating Sample C# Web Applications, ASP.NET Web Page Execution Architecture, Debugging and Tracing of ASP.NET.

UNIT V

ASP.NET WEB CONTROLS: Web Form Structures, Introduction to Web Form controls, Server Side Controls, Web Server Controls, GET and POST, Page Submission, Web Page Creation Techniques, Redirection between Web Pages, Validation Controls.

UNIT VI

WEB SERVICES: Web Services, Web Service Architecture, WSDL, Building WSDL Web Service.

CONFIGURING .NET ASSEMBLIES: Private Assemblies, Shared Assemblies, and Versioning.

MULTITHREADED PROGRAMMING: Thread Class, Life Cycle of a Thread, Steps for Creating a Thread, Thread Synchronization.

TEXT BOOKS:

- Pro C# 2010 and the .NET 4 Platform, Andrew Trolesen,
- Programming in C#, E. Balagurusamy, Tata McGraw-Hill, 2004.

REFERENCE BOOKS:

- Programming C#, J. Liberty, 2nd Edition., O'Reilly, 2002.
- C# and the .NET Platform, Andrew Trolesen, 2nd Edition, Dreamtech Press
- Sams Teach Yourself the C# Language in 21 Days', Bradley L Jones, 1st edition, 2001.
- Microsoft C# Programming for the Absolute Beginner, Andy Harris, PTR publications, 2002.
- The Complete Reference: C#, Herbert Schildt, Tata McGraw-Hill, 2004.
- Professional C#, Robinson et al, 2nd Edition., Wrox Press, 2002.
- C# and the .NET Platform, Andrew Troelsen, A! Press, 2003.
- A Textbook on C#, S. ThamaraiSelvi, R. Murugesan, Pearson Education, 2003.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

III B.Tech. I-Sem (CSE)

T	C
3	2

(A0013125) PROFESSIONAL ETHICS AND SOFT SKILLS

(Soft Skill Development Course)

(Common to all branches)

OBJECTIVES

The recent past decades have witnessed a dilemma of performance on ethical grounds. A professional be able to carry out tasks and achieve success at societal level. The syllabus has been designed keeping in view of the needs and goals of the generation next undergraduates. It comprises essentials of professional ethics embedded with soft skills which in turn mould students as dynamic professionals. The course of **Professional Ethics and Soft Skills** has been designed with the following objectives.

- To ignite the spark of professionalism among students with the purpose to acquire success at societal level.
- To enable them to accomplish tasks balancing hard skills and soft skills.
- To develop critical thinking skills and emotions of students through recent research theories.
- The greatest contribution of this course shall be to shape human skills of students at the global level.

OUTCOMES

- Be able to acquire professional ethics & Job Etiquettes
- Be able to balance hard skills and soft skills.
- Considerable improvement in communicative ability.
- Increase in motivational level and Professional attitudes.
- Be able to possess wide range of relevant knowledge.

UNIT I

NATURE AND SCOPE OF ENGINEERING ETHICS: Definition, Nature, Scope – Moral Dilemmas – Moral Autonomy – Kohlberg’s theory – Gilligan’s theory, the characteristic traits of real professional, Moral Reasoning and Ethical theories – Theories of Right Action, Self – interest- Use of ethical Theories- case study.

UNIT II

PROFESSIONAL ETIQUETTES: Professional Etiquettes – Mobile Etiquettes – Email Etiquettes -Kinesics – Proxemics - Chronemics – Chromatics – Olfacts - Hap tics – Case Study.

UNIT III

CORPORATE COMMUNICATION: Communication models- Types of Communication – downward and upward communication Business Deliberations – Meetings – Negotiation Skills - Case Study.

UNIT IV

SOFT SKILLS: Interpersonal Communication – Johari Window – Interpersonal conflict resolutions- Daniel Goleman’s Emotional Intelligence.

UNIT V

GLOBAL ISSUES: Multinational corporations – cross-cultural communication-Environmental ethics – Computer – ethics –Terrel Ward Bynum’s concept of computer ethics - Weapons developments- case study.

UNIT VI

INTRODUCTION TO INTELLECTUAL PROPERTY: Meaning and Types of Intellectual Property – recent developments of the copy right act – plagiarism – trademark protection – patent law.

TEXT BOOKS:

1. Charles D.Fleddermann [1999], *Engineering Ethics*, Prentice Hall Publishers, New Mexico.
2. Business Communication , P.D. Chaturvedi, Mukesh Chaturvedi

REFERENCES :

1. The ACE of Soft Skills(Attitude, Communication and Etiquette for success) by – Gopaldaswamy Ramesh & Mahadevan Ramesh, Pearson 2010.
2. Essentials of Business Communication, Rajendra Pal, JS.Korlahhi, S.Chand
3. Intellectual Property Right , Deborah E. BouchouxS, Cengage, 2005
4. Business Ethics and Professional Values, A.B. Rao, Excel,2009
5. M.P. Raghavan [2006], Professional Ethics And Human Values, Scitech Publications, Chennai.

(A0597125) ADVANCED JAVA PROGRAMMING LAB**Objective :**

- To create a fully functional website with mvc architecture. To Develop an online Book store using we can sell books (Ex amazon .com).

Hardware and Software required :

- A working computer system with either Windows or Linux
 - A web browser either IE or firefox
 - Tomcat web server and Apache web server
 - XML editor like Altova Xml-spy [www.Altova.com/XMLSpy – free] , Stylusstudio , etc.,
 - A database either Mysql or Oracle
 - JVM(Java virtual machine) must be installed on your system
 - BDK(Bean development kit) must be also be installed
- Develop an applet in Java that displays a simple message.
 - Develop an applet in Java that receives an integer in one text field, and computes its factorial Value and returns it in another text field, when the button named “Compute” is clicked.
 - Write a java program to draw ovals, circles, rectangle, rounded rectangles & arcs using applets.
 - Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -, *, % operations. Add a text field to display the result.
 - Write a Java program for handling mouse events.
 - Write a Java program for handling key events.
 - Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception Display the exception in a message dialog box.
 - Write a Java program that implements a simple client/server application. The client sends data to a server. The server receives the data, uses it to produce a result, and then sends the result back to the client. The client displays the result on the console. For ex: The data sent from the client is the radius of a circle, and the result produced by the server is the area of the circle. (Use java.net)
 - Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green. When a radio button is selected, the light is turned on, and only one light can be on at a time No light is on when the program starts.
 - Write a Java program that allows the user to draw lines, rectangles and ovals.
 - Write a java program to develop grid layout & border layout. When user clicks the button background color has to change according to given color.
 - Write a java program to develop JMenu.
 - Write a java program to develop JList and JComboBox.
 - Create a simple visual bean with a area filled with a color. The shape of the area depends on the property shape. If it is set to true then the shape of the area is Square and it is Circle, if it is false. The color of the area should be changed dynamically for every mouse click. The color should also be changed if we change the color in the “property window “.
 - Create a bean that performs conversion of American dollar to Indian rupee.
 - Install Tomcat web server.
 - Write a simple servlet and run it using tomcat server.
 - Write a servlet to read parameters from client request.

(A0598125) C# AND .NET FRAMEWORK LAB

(Common to CSE & IT)

LAB-1

1. Write a program to C# to find the smallest single digit factor for a given value.
2. Write a program in C# to print a number if it is prime; otherwise display the largest factor of that number.
3. Write a program in C# to find the magnitude of a number.

LAB-2

4. Write a C# program for addition and multiplication of two matrices.
5. Write a C# program to display the digits of an integer in words.
6. Write a C# program to which reads a set of strings into the rows a two dimensional array and then prints the string having more number of vowels..

LAB-3

7. Write a C# programs to demonstrate the concepts of Structures and Enumerations.
8. Write a C# programs to demonstrate the concepts of Constructors and Inheritance.
9. Write a C# programs to demonstrate the concepts of Polymorphism.

LAB-4

10. Write a C# programs to demonstrate the concepts of Partial classes and Extension methods.
11. Write a C# programs to demonstrate the concepts of Delegates.

LAB-5

12. Write a C# programs to demonstrate the concepts of Label, Text Box and Button controls.
13. Write a C# programs to demonstrate the concepts of Combo Box and List Box controls.

LAB-6

14. Create a Windows application in C# for registration form and fill the details and when you click the submit button it display the details in the message box.
15. Create a Windows application in C# having two text boxes and three buttons named as factorial, prime, factorial series. When you click any button the resultant value will be displayed on the second textbox.

LAB-7

16. Create a ADO.NET application in C# to verify if the connection is established with OLEDB and MS-ACCESS.
17. Create a ADO.NET applications in C# to demonstrate the Data Reader, Data Set, Data Adapter and Data View Objects.

LAB-8

18. Develop the Static Web pages using HTML and some validations along with Java Script.
19. Design an ASP.NET Webpage to demonstrate the Label, Button and Textbox controls.

LAB-9

20. Design an ASP.NET Webpage to work with Cross page and Post back Submissions.
21. Design an ASP.NET Webpage to work with Dropdown list and ListBox controls.

LAB-10

22. Develop a Registration Form with all Validation Controls.
23. Create a Web Service for all Arithmetic operations.

LAB-11

24. Write a C# program to implement Assemblies.
25. Write a C# program to implement Multi Threading and Thread Synchronization.

(A0599125) COMPUTER NETWORKS & COMPILER DESIGN LAB**OBJECTIVE:**

1. To understand the functionalities of various layers of OSI model
2. To understand the operating System functionalities

COMPUTER NETWORKS

1. Implement the data link layer framing methods such as character, characterstuffing and bit stuffing.
2. Implement on a data set of characters the three CRC polynomials – CRC 12, CRC 16 and CRC CCIP.
3. Implement Dijkstra's algorithm to compute the Shortest path thru a graph.
4. Take an example subnet graph with weights indicating delay between nodes.
5. Now obtain Routing table at each node using distance vector routing algorithm.
6. Take an example subnet of hosts. Obtain broadcast tree for it.

COMPILER DESIGN:

1. Write a program to check whether a given string belongs to the grammar or not
2. Write a program to generate a parse tree
3. Write a program to find leading terminals
4. Write a program to find trailing terminals
5. Write a program to compute FIRST of non terminals
6. Write a program to compute FOLLOW of non terminals

(A0517126) MOBILE COMPUTING

(Common to CSE&IT)

Course Objectives:

- Introduction of an advanced element of learning in the field of wireless communication
- introduces the basic concepts and principles in mobile computing
- Expose the students to the concepts of wireless devices and mobile computing.

Course Outcomes:

At the end of this course the student should be able to

- Students to understand the concept of mobile computing and architecture of mobile communication.
- Students to apply the concepts of mobile communications to the transactions and transaction management
- Apply the concepts of mobile computing on conventional wired networks
- Students to understand the working of heterogeneous networks.

Prerequisites:

- The knowledge about the spectrum and information the radio waves.
- The knowledge of Computer networks and protocols is essential.

UNIT I**Introduction to Mobile Communication and Computing:** Novel applications – Limitations - Architecture**GSM:** Mobile services, System architecture, Radio Interface, Protocols, Localization and calling, Handover, Security, New data Services**UNIT II****(Wireless) Medium Access Control:** Motivation for a specialized MAC (Hidden and exposed terminals, Near and far terminals), SDMA, FDMA, TDMA, CDMA.**UNIT III****Mobile Network Layer:** Mobile IP (Goals, assumptions, entities and terminology, IP packet delivery, agent advertisement and discovery, tunnelling and encapsulation), Dynamic Host Configuration Protocol (DHCP), Properties and Applications of a MANETs - Routing – DSDV – DSR**UNIT IV****Mobile Transport and Application Layer:** Traditional TCP - Indirect TCP - Snooping TCP - Mobile TCP – Fast retransmit/Fast recovery - Transmission/time-out freezing - Selective retransmission – Transaction Oriented TCP – Wireless Application Protocol**UNIT V****Database Issues:** Hoarding techniques - caching invalidation mechanisms - client server computing with adaptation - power-aware and context-aware computing - transactional models - query processing – recovery - Quality of service issues.**UNIT VI****Data Dissemination:** Communications asymmetry - classification of new data delivery mechanisms - push-based mechanisms - pull-based mechanisms - hybrid mechanisms - selective tuning (indexing) techniques.**TEXT BOOKS :**

1. Jochen Schiller, “Mobile Communications”, *Addison-Wesley*. (Chapters 4,7,9,10,11),second edition, 2004.
2. Stojmenovic and Cacute, “Handbook of Wireless Networks and Mobile Computing”, *Wiley*, 2002.

REFERENCES:

1. Reza Behravanfar, “Mobile Computing Principles: Designing and Developing Mobile Applications with UML and XML”, ISBN: 0521817331, Cambridge University Press, October 2004.
2. Adelstein, Frank, Gupta, Sandeep KS, Richard III, Golden, Schwiebert, Loren, “Fundamentals of Mobile and Pervasive Computing”, ISBN: 0071412379, McGraw-Hill Professional, 2005.
3. Hansmann, Merk, Nicklous, Stober, “Principles of Mobile Computing”, *Springer*, second edition, 2003.
4. MartynMallick, “Mobile and Wireless Design Essentials”, *Wiley DreamTech*, 2003.

(A0518126) DESIGN AND ANALYSIS OF ALGORITHMS**OBJECTIVES:**

- Significance of algorithms in the computer field.
- Various aspects of algorithm development.
- Qualities of a good solution.
- Significance and importance of program correctness.
- Various fields that constitute the subject of algorithms.
- Relationship between algorithms and other fields in the computer realm.
- How the field of algorithms are developed.
- Significance of algorithm efficiency.

OUTCOMES:

- Explain the basic concepts of time and space complexity, divide-and-conquer strategy, dynamic programming, greedy algorithms, amortized analysis.
- Describe the methodologies of how to analyze an algorithm.
- Identify the complexity of problems.
- Solve a problem using an algorithm and evaluate its correctness.
- Formulate the time-complexity analysis for an algorithm.
- Design a better algorithm to solve the problems.
- Build up analyzing, designing and programming skills

UNIT I

Introduction: Algorithm, Pseudo code for expressing algorithms, Performance Analysis-Space complexity, Time complexity, Asymptotic Notation- Big oh notation, Omega notation, Theta notation and Little oh notation, Probabilistic analysis, Amortized analysis.

UNIT II

Divide and conquer: General method , applications-Binary search, Finding Maximum and minimum ,Quick sort, Merge sort, Strassen's matrix multiplication.

UNIT III

Greedy method: General method, applications-Job sequencing with dead lines, knapsack problem, Minimum cost spanning trees, Single source shortest path problem.

UNIT IV

Dynamic Programming: General method, applications- 0/1 knapsack problem, All pairs shortest path problem, Travelling sales person problem, Reliability design, optimal binary search tree.

UNIT V

Backtracking: General method, applications-n-queen problem, sum of subsets problem, graph coloring.
Branch and Bound: General method, applications - Travelling sales person problem.

UNIT VI

NP-Hard and NP-Complete problems: Basic Concepts, Non Deterministic algorithms, the classes of NP Hard and NP Complete, Cook's Theorem.

TEXT BOOKS:

1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rjasekharam, Galgotia publications pvt. Ltd.
2. Algorithm Design: Foundations, Analysis and Internet examples, M.T.Goodrich and R.Tomassia, John wiley and sons.

REFERENCES:

1. Introduction to Algorithms, second edition, T.H.Cormen, C.E.Leiserson, R.L.Rivest,andC.Stein,PHI Pvt. Ltd./ Pearson Education.
2. Introduction to Design and Analysis of Algorithms A strategic approach, R.C.T.Lee, S.S.Tseng, R.C.Chang and T.Tsai, Mc Graw Hill.
3. Data structures and Algorithm Analysis in C++, Allen Weiss, Second edition, Pearson education.
4. Design and Analysis of algorithms, Aho, Ullman and Hopcroft, Pearson education.
Algorithms – Richard Johnson baugh and Marcus Schaefer, Pearson Education.

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

III B.Tech. II-Sem (CSE)

T C

3+1* 3

(A1229126) INFORMATION SECURITY

(Common to CSE & IT)

Course Objectives:

- This course covers the major aspects of computer and network security. It starts with a general introduction to information security, then proceeds to cover types of threats and attacks, hacking techniques, network vulnerabilities, security policies and standards, firewalls, cryptography, Authentication & digital signatures, the SSL protocol, Wireless security, intrusion detection and prevention

Learning Outcomes:

The student will be able to:

- Define information security and outline its major components.
- Identify the major types of threats to information security and the associated attacks.
- Develop strategies to protect organization information assets from common attacks.
- Understand how security policies, standards and practices are developed.
- Understand the role of management in enforcing security policies, standards and practices.
- Identify the major techniques, approaches and tools used to discover network and system vulnerabilities.
- Understand firewalls and packet filtering.

UNIT - I

Security goals, Security Attacks (Interruption, Interception, Modification and Fabrication), Security Services (Confidentiality, Authentication, Integrity, Non-repudiation, access Control and Availability) and Mechanisms, A model for Inter network security, Internet Standards and RFCs, Buffer overflow & format string vulnerabilities, ARP attacks, route table modification and man-in-the-middle attacks.

UNIT - II

Conventional Encryption Principles, Conventional encryption algorithms, cipher block modes of operation, location of encryption devices, key distribution Approaches of Message Authentication, Secure Hash Functions and HMAC.

UNIT - III

Public key cryptography principles, public key cryptography algorithms, digital signatures, digital Certificates, Certificate Authority and key management Kerberos, X.509 Directory Authentication Service.

UNIT - IV

Email privacy: Pretty Good Privacy (PGP) and S/MIME.

UNIT - V

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

UNIT - VI

Intruders, Viruses and related threats. Firewall Design principles, Intrusion Detection Systems.

TEXT BOOKS :

1. Network Security Essentials (Applications and Standards) by William Stallings Pearson Education.
2. Hack Proofing your network by Ryan Russell, Dan Kaminsky, Rain Forest Puppy, Joe Grand, David Ahmad, Hal Flynn Ido Dubrawsky, Steve W.Manzuik and Ryan Permeh, wiley Dreamtech

REFERENCES :

1. Fundamentals of Network Security by Eric Maiwald (Dreamtech press)
2. Network Security - Private Communication in a Public World by Charlie Kaufman, Radia Perlman and Mike Speciner, Pearson/PHI.
3. Cryptography and network Security, Third edition, Stallings, PHI/Pearson.
4. Principles of Information Security, Whitman, Thomson.
5. Network Security: The complete reference, Robert Bragg, Mark Rhodes, TMH.
6. Introduction to Cryptography, Buchmann, Springer.

(A0520126) SOFTWARE TESTING METHODOLOGIES AND TOOLS

(Common to CSE & IT)

Background:

Software testing is an integral and important activity in every software development environment. Software seems to have permeated almost every equipment that we use in our daily lives.

This course is designed to enable a clear understanding and knowledge of the foundations, techniques, and tools in the area of software testing and its practice in the industry. The course will prepare students to be leaders in software testing. Whether you are a developer or a tester, you must test software. This course is a unique opportunity to learn strengths and weaknesses of a variety of software testing techniques.

COURSE OBJECTIVES:

Upon successful completion of this course students will be able to:

1. Understand the basic concepts of software testing.
2. Understand the various techniques and strategies of software testing and inspection and pointing out the importance of testing in achieving high-quality software.
3. Perform effective and efficient structural testing of software.
4. Integrate and test the various units and components of a software system.
5. Perform effective and efficient functional testing of software.
6. Select the appropriate tests to regression test your software after changes have been made.
7. Plan, track and control the software testing effort.
8. Understand the need of automated testing tools and various kinds of automated testing tools.

LEARNING OUTCOMES:

1. Have an ability to apply software testing knowledge and engineering methods.
2. Have an ability to design and conduct a software test process for a software testing project.
3. Have an ability to identify the needs of software test automation, and define and develop a test tool to support test automation.
4. Have an ability understand and identify various software testing problems, and solve these problems by designing and selecting software test models, criteria, strategies, and methods.

UNIT I :

Testing Methodology: Introduction to software Testing, Evolution of Software Testing, Goals of Software Testing, Software testing as a Process. Software Testing Terminology, Software Testing Life Cycle. Verification & Validation: Verification and Validation Activities, Verification and its requirements, Validation.

UNIT II:

Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs.

UNIT III:

Flow graphs and Path testing: Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

UNIT IV:

Dataflow testing:-Basics of dataflow testing, strategies in dataflow testing, application of dataflow testing.

UNIT V:

Paths, Path products and Regular expressions: path products & path expression, reduction Procedure, applications, regular expressions & flow anomaly detection.

Graph Matrices and Application: Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm.

UNIT VI:

Test Automation: Need for Automation, Categorization of Testing Tools, Overview of Some Commercial Testing Tools: Win Runner, Load Runner, QTP, Selenium

TEXT BOOKS:

1. Software testing techniques - Boris Beizer, 2nd Edition, Dreamtech.
2. Software Testing Principles and Practices by NARESH CHAUHAN, OXFORD HIGHER EDUCATION, university Press, 2013.

REFERENCE BOOKS:

1. Software Testing in the Real World – Edward Kit, Pearson.
2. Effective methods of Software Testing, Perry, John Wiley.
3. Art of Software Testing – Meyers, John Wiley.
4. Software testing Tools – Dr.K.V.K.K.Prasad, Dreamtech.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

III B.Tech. II-Sem (CSE)

T C
3+1* 3**(A0521126) UNIX AND SHELL PROGRAMMING**

(Common to CSE & ECE)

OBJECTIVES:

Upon successful completion of this course, you should be able to:

- Define and redirect standard files.
- Use the pipe operator to connect two commands.
- Use wildcard met characters and the history command to recall commands.
- Use shell variables.
- Define and use foreground and background processes.
- Define and use processes and co-processes.
- Use directory information in scripts.
- Set and use positional parameters and escape sequences.
- Pass arguments to shell scripts and test arguments.
- Use conditions, control statements and the if command in a decision.
- Use while and for loops.
- Debug scripts using execute trace and verbose trace.
- Use command substitution and Group shell commands.
- Create and use aliases and functions.
- Identify and use signals and handle signals in a shell script.
- Use integer and floating-point arithmetic.
- Create and use variable arrays.
- Perform command evaluation using eval.
- Use here documents, file I/O operators and command options processing.
- Use a lock file to synchronize access.

OUTCOMES:

- Be familiar with Unix and Linux operating Systems.
- Master the techniques to use a Linux system.
- Be familiar with the Unix file system and its basic operations.
- Be familiar with the Unix command interpreters.
- Master the techniques of shell programming.

UNIT I

Introduction: Why Unix?, Computer System, The Unix Environment, Unix structure, Accessing Unix, Common commands: date, cal, who, password, echo, man, lpr. Other useful commands: tty, clear, sty, script, uname, bc, tar, **Vi editor**: Editor concepts, The vi editor, Modes, Commands.

UNIT II

File Systems: File Names, File Types, Regular Files, Directories, File System Implementation, Operations unique to regular files, Operations unique to directories, Operations common to both.

Security & File Permissions: users and groups, security levels, changing permissions, user masks, changing ownership and group.

UNIT III

Introduction to Shells: Unix Session, Standard Streams, Redirection, Pipes, Tee Command, Command Execution, Command-Line Editing, Quotes, Command Substitution, Job Control, Aliases, Variables, Predefined Variables, Options, Shell/Environment Customization.

UNIT IV

Filters: Filters and Pipes, Concatenating files, Display Beginning and End of files, Cut and Paste, Sorting, Translating Characters, Files with Duplicate Lines, Count characters, Words or Lines, Comparing Files.

Grep: Operation, grep Family, Searching for File Content.

UNIT V

awk: Execution, Fields and Records, Scripts, Operations, Patterns, Actions, Associative Arrays, String Functions, String Functions, Mathematical Functions, User – Defined Functions, Using System commands in awk, Applications, awk and grep.

UNIT VI

Interactive Korn Shell: Korn Shell Features, Two Special Files, Variables, Output, Input, Exit Status of a Command, eval command, Command Execution Process.

Korn Shell Programming: Basic Script concepts, Expressions, special Parameters and Variables, changing Positional Parameters, Argument Validation, Script Examples.

TEXT BOOKS:

1. Unix and shell Programming Behrouz A. Forouzan, Richard F. Gilberg. Thomson
2. Your Unix the ultimate guide, Sumitabha Das, TMH. 2nd Edition.

(A0522126) CLOUD COMPUTING
(ELECTIVE - I)**OBJECTIVES:**

- The objective focuses on technologies specific to the networked, distributed dimension of software and access to services and data. It will support long-term research on new principles, methods, tools and techniques enabling software developers in the EU to easily create interoperable services based on open standards, with sufficient flexibility and at a reasonable cost.

OUTCOMES:

- Intelligent and autonomic management of cloud resources, ensuring agile elastic scalability. Scalable data management strategies, addressing the issues of heterogeneity, consistency, availability, privacy and supporting security.
- Technologies for infrastructure virtualization, cross platforms execution as needed for service composition across multiple, heterogeneous environments, autonomous
- Management of hardware and software resources.
- Interoperability amongst different clouds, portability, protection of data in cloud environments, control of data distribution and latency.
- Seamless support of mobile, context-aware applications.

UNIT 1

Examining the Value Proposition: Defining Cloud Computing [CC for short]: Definition; Cloud types; characteristics of CC; Role of open standards. Assessing the Value Proposition: Measuring the cloud value, Avoiding capital expenditure, Total cost of ownership, Specifying SLAs.

UNIT 2

Architecture and types: Cloud Architecture: CC stack; Connecting to the cloud.
Services and Applications by type: IaaS, PaaS, SaaS, IDaaS, CaaS.

UNIT 3

Platforms: Abstractions and Virtualization: Using virtualization technologies; Load balancing and virtualization, Hypervisor, Machine image, porting applications.

Capacity planning: defining baseline and metrics; Network capacity.

Platform as a Service: Defining service; PaaS application frameworks;

UNIT 4

Cloud computing with Titans: Google and Amazon web services, Microsoft cloud services.

UNIT 5

Exploring Cloud Infrastructures: Managing the cloud: Administrating the cloud; Cloud management products; Emerging cloud management standards.

Cloud Security: Securing the cloud; securing data; Establishing identity and presence.

UNIT 6

Services & Applications: Service Oriented Architecture: Introduction; Defining SOA communications; Managing and monitoring SOA; Relating SOA and CC. Moving applications to cloud. Cloud based storage. Webmail services.

TEXT BOOKS:

1. Cloud Computing – Bible, Barrie Sosinsky, Wiley-India edn.,2012. [Chapters 1-7, 11-15, 17.]
2. Cloud Computing – A Practical Approach, Anthony T. Velte, Toby J. Velte and Robert Elsenpeter, Tata McGraw-Hill edn. 2011. [Chapter3].

(A0523126) DISTRIBUTED SYSTEMS
(ELECTIVE-I)**OBJECTIVES:**

- Present the principles underlying the functioning of distributed systems;
- Create an awareness of the major technical challenges in distributed systems design and implementation;
- Expose students to modern and classic technology used in distributed systems and their software;
- Provide experience in the implementation of typical algorithms used in distributed systems

OUTCOMES:

After completing this course the student will be able to:

- Explain what a distributed system is, why you would design a system as a distributed system, and what the desired properties of such systems are;
- List the principles underlying the functioning of distributed systems, describe the problems and challenges associated with these principles, and evaluate the effectiveness and shortcomings of their solutions;
- Recognise how the principles are applied in contemporary distributed systems, explain how they affect the software design, and be able to identify features and design decisions that may cause problems;
- Design a distributed system that fulfils requirements with regards to key distributed systems properties (such as scalability, transparency, etc.), be able to recognise when this is not possible, and explain why;

UNIT I

INTRODUCTION: What is Distributed System? Goals: Advantages of Distributed Systems over Centralized Systems, Advantages of Distributed Systems over Independent PCs, Disadvantages of Distributed Systems, Hardware Concepts, Design Issues: Transparency, Flexibility, Reliability, Performance, Scalability.

Communication in Distributed Systems I: Layered Protocols.

UNIT II

Communication in Distributed Systems II: The Client Server Model: Clients and servers, Addressing, Blocking Versus Nonblocking Primitives, Buffered versus Unbuffered Primitives, Reliable versus Unreliable Primitives, Implementing the Client-Server Model, Remote Procedure call, Group Communication.

UNIT III

Synchronization in Distributed Systems: Clock synchronization, Mutual Exclusion: A centralized Algorithm, A Distributed Algorithm, A Token Ring Algorithm, A comparison of three algorithms, Election Algorithms: The Bully Algorithm, A Ring Algorithm, Atomic Transactions, Deadlocks in Distributed systems.

UNIT IV

Processes and Processors in Distributed Systems: Threads, System Models, Processor Allocation, Scheduling in Distributed Systems, Fault Tolerance, Real-Time Distributed Systems: What is a Real-Time System? Design Issues, Real-Time Communication, Real-Time Scheduling.

UNIT V

Distributed File Systems and Shared Memory: Distributed File System Design: The File Service Interface, The Directory Service Interface, Semantics of File Sharing, Distributed File System Implementation: File Usage, System Structure, Caching, Replication, What is Shared Memory? Page-Based Distributed Shared Memory.

UNIT VI

CASE STUDY 1 AMOEBA: Introduction to AMOEBA, Objects and Capabilities, Process Management, Memory Management, Communication.

CASE STUDY 2 MACH: Introduction to MACH, Process management, Memory Management, Communication.

TEXT BOOKS:

1. Distributed Operating Systems, A.S.Tanenbaum, Pearson Education.

REFERENCES:

1. Distributed Systems Concepts and Design, G Coulouris, J Dollimore and T Kindberg, Fourth Edition, Pearson Education
2. Distributed Operating Systems Concepts and Design, Pradeep K.Sinha, PHI.
3. Advanced Concepts in Operating Systems, M Singhal, N G Shivarathri, Tata McGraw-Hill Edition.
4. Reliable Distributed Systems, K.P.Birman, Springer.
5. Distributed Operating Systems and Algorithm Analysis, R.Chow, T.Johnson, Pearson.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

III B.Tech. II-Sem (CSE)

T C
3+1* 3**(A0524126) ADVANCED COMPUTER ARCHITECTURE
(ELECTIVE-I)****OBJECTIVES:**

- A broad understanding of computer architecture.
- To the extent possible, an understanding of the current state-of-the-art in uni-processor computer architecture.
- Study how to use technology to build the best computer/processor.
- To know different levels of parallelism.
- Issues in interconnection of networks and cluster design.

OUTCOMES:

Upon completion of this course, a student should have developed:

- Broad understanding of the design of computer systems, including modern architectures and alternatives.
- Understanding of the interaction amongst architecture, applications and technology.
- Understanding of a framework for evaluating design decisions in terms of application requirements and performance measurements.
- A historical perspective on computer system design.

UNIT -I**Fundamentals of computer design**-Technology Trends, Cost , Measuring and reporting performance, Quantitative principles of computer design.**UNIT -II****Instruction set principle and examples**-Classifying instruction set, Memory addressing, Addressing modes for signal processing, Operations in the instruction set, Instructions for control flow, encoding an instruction set, the role of the compiler.**UNIT -III****Instruction level parallelism**-Over coming data hazards, Reducing branch costs, High performance instruction delivery, Hardware based speculation, Compiler techniques, Static branch prediction, H.W. vs S.W solutions, Limitations of ILP**UNIT -IV****Memory hierarchy design**-Cache performance, Reducing cache miss penalty and miss rate, Virtual memory, Protection and examples of virtual memory.**UNIT -V****Multi processors and thread-level parallelism**-symmetric shared memory architectures , Distributed -shared memory architectures, Synchronization, Multi-threading .**UNIT -VI****Storage systems** Types of Storage Devices, RAID, Errors and failures in real time systems,**Interconnection Networks-** Interconnection network media, Practical issues for commercial interconnection of networks, Clusters, Designing a cluster.**TEXT BOOK:**

1. Computer Architecture A quantitative approach 3rd edition John L. Hennessy & David A. Patterson Morgan Kufmann (An Imprint of Elsevier)

REFERENCES:

1. "Computer Architecture and parallel Processing" Kai Hwang and A.Briggs International Edition McGraw-Hill.
2. Advanced Computer Architectures, Dezso Sima, Terence Fountain, Peter Kacsuk, Pearson.

(A0525126) VIRTUAL REALITY
(ELECTIVE-I)**OBJECTIVES:**

- The course objective is to promote the understanding of this technology, underlying principles, its potential and limits and to learn about the criteria for defining useful applications. Furthermore, students will be exposed to the process of creating virtual environments, by developing a complete VR application as members of a small team. Project teams will include students from different disciplines, thereby, complementing technical skills with imagination, creativity, and innovative ideas.

OUTCOMES:

On completion of the course the student will:

- Have skills in building product models by utilizing technologies for virtual environments.
- Be able to describe possible application fields of Virtual Reality in industry.
- Have insight in on-going research activities in this particular field.

UNIT-I

Introduction: The three I's of virtual reality, commercial VR technology and the five classic components of a VR system.

UNIT - II

Input Devices: (Trackers, Navigation, and Gesture Interfaces): Three-dimensional position trackers, navigation and manipulation, interfaces and gesture interfaces.

UNIT - III

Output Devices: Graphics displays, sound displays & haptic feedback.

UNIT - IV

Modelling: Geometric modelling, kinematics modelling, physical modelling, behaviour modelling, model management.

UNIT - V

Human Factors: Methodology and terminology, user performance studies, VR health and safety issues.

UNIT - VI

Applications: Medical applications, military applications, robotics applications.

TEXT BOOKS :

1. Virtual Reality Technology, Second Edition, Gregory C. Burdea & Philippe Coiffet, John Wiley & Sons, Inc.,
2. Killer Game Programming in Java, Andrew Davison, Oreilly-SPD, 2005.

REFERENCES:

1. Understanding Virtual Reality, interface, Application and Design, William R.Sherman, Alan Craig, Elsevier(Morgan Kaufmann).
2. 3D Modeling and surfacing, Bill Fleming, Elsevier(Morgan Kauffman).
3. 3D Game Engine Design, David H.Eberly, Elsevier.
4. Virtual Reality Systems, John Vince, Pearson Education.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

III B.Tech. II-Sem (CSE)

T	C
3	2

(A0526126) INTRODUCTION TO ORACLE 9i
(SKILL DEVELOPMENT COURSE)

OBJECTIVES

Student will able to:

- Create and delete database schemas and execute SQL queries
- Inserting data, Altering and dropping the tables.
- Various types of data conversions using the functions.
- This course introduces students to data server technology and explores the concepts of relational databases, the SQL*Plus interface tool, and the powerful SQL programming languages.
- Students will learn and practice how to create and maintain database objects.
- The students will also learn to store, retrieve, and manipulate data.
- This course will help the student to prepare for Oracle's exam, Introduction to Oracle9i SQL

OUTCOMES

Upon completion of the lab, the student should be able to:

- Map the model into a relational database system.
- Implement the given schema on a relational DBMS.
- Design, develop, and maintain Oracle Database Objects.
- Use a database language for manipulating and querying data.
- Develop advanced packages, stored procedures, and triggers and functions using PL/SQL
- The learning objectives of this course are to produce knowledge and highly skilled graduates in the field of information and communication technology.
- Graduates pursuing the program are equipped with the in-depth knowledge and specialized skills in database management area.
- This includes the ability to analyse, design, develop program using structured programming method, manage and maintain database system which could meet the industrial needs in the field.
- Students should be able to develop data mining application with required security standard to protect the system database.

UNIT –I

Introduction, Writing Basic SQL SELECT statements

UNIT –II

Restricting and Sorting Data, Single row functions

UNIT –III

Displaying Data From Multiple Tables, Aggregating Data Using Group Functions

UNIT-IV

Sub queries, Manipulating Data

UNIT-V

Creating and Managing Tables

UNIT –VI

Including Constraints

REFERENCES:

1. Oracle University Press

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

III B. Tech. II- Sem (CSE)

P	C
3	2

(A0581126) SOFTWARE TESTING TOOLS LAB

(Common to CSE & IT)

COURSE OBJECTIVES:

Upon successful completion of this course students will be able to:

- Understand the basic concepts of software testing.
- Understand the various techniques and strategies of software testing and inspection and pointing out the importance of testing in achieving high-quality software.
- Perform effective and efficient structural testing of software.
- Integrate and test the various units and components of a software system.
- Perform effective and efficient functional testing of software.
- Select the appropriate tests to regression test your software after changes have been made.
- Plan, track and control the software testing effort.
- Understand the need of automated testing tools and various kinds of automated testing tools.

LEARNING OUTCOMES:

- Have an ability to apply software testing knowledge and engineering methods.
- Have an ability to design and conduct a software test process for a software testing project.
- Have an ability to identify the needs of software test automation, and define and develop a test tool to support test automation.
- Have an ability understand and identify various software testing problems, and solve these problems by designing and selecting software test models, criteria, strategies, and methods.

Lab Experiments:

1. Write programs in 'C' Language to demonstrate the working of the following constructs:
2. i) do...while
ii) while....do
iii) if...else
iv) switch
v) for
3. "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 manual test cases for Gmail application.
5. Write manual test cases for ATM application.
6. Write manual test cases for Banking application.
7. Study of Quick Test Professional(QTP):
8. Overview of QTP Components.
9. Record & Run Options.
10. Generating Basic Script.
11. Enhancement of Script.
12. Check Points.
13. Output Values.
14. Object Repository.
15. Writing Script manually.
16. Study of Rational Functional Tester(RFT).
17. Study of SELENIUM.

Note: this lab deals only with testing tools and not case tools as case tools are already covered for IT students in the II year I semester in 'Fundamentals of OOD'

(A1294124) UNIX AND SHELL PROGRAMMING LAB

(Common to CSE & IT)

OBJECTIVES:

- To teach students various unix utilities and shell scripting.

OUTCOMES

- Mastery of the basic UNIX process structure and the UNIX file system.
- Mastery of simple UNIX filters.
- Familiarity with pipes and redirection. the UNIX environment, traps.
- Signals, filter parameters, filter options, UNIX contentions, and Regular Expressions.

Recommended Systems/Software Requirements:

- Intel based desktop PC with minimum of 166 MHZ or faster processor with at least 64 MB RAM and 100 MB free disk space LAN Connected
- Any flavour of Unix / Linux

Week1**SESSION-1**

- Log into the system.
- Use vi editor to create a file called myfile.txt which contains some text.
- Correct typing errors during creation.
- Save the file e)logout of the system

SESSION-2

- Log into the system
- Open the file created in session 1
- Add some text
- Change some text
- Delete some text.
- Save the Changes
- Logout of the system

Week2

- Log into the system
- Use the cat command to create a file containing the following data. Call it mytable use tabs to separate the fields.

1425	Ravi	15.65
4320	Ramu	26.27
6830	Sita	36.15
1450	Raju	21.86
- Use the cat command to display the file, mytable.
- Use the vi command to correct any errors in the file, mytable.
- Use the sort command to sort the file mytable according to the first field. Call the sorted file my table (same name)
- Print the file mytable
- Use the cut and paste commands to swap fields 2 and 3 of mytable. Call it my table (same name)
- Print the new file, mytable
- Logout of the system.

Week3

- Login to the system
- Use the appropriate command to determine your login shell
- Use the /etc/passwd file to verify the result of step b.
- Use the who command and redirect the result to a file called myfile1. Use the more command to see the contents of myfile1.
- Use the date and who commands in sequence (in one line) such that the output of date will display on the screen and the output of who will be redirected to a file called myfile2. Use the more command to check the contents of myfile2.

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**Week4**

- a) Develop an interactive grep script that asks for a word and a file name and then tells how many lines contain that word.
- b) Repeat
- c) Part using awk.

Week5

- a) Write a shell script that takes a command –line argument and reports on whether it is directory, a file, or something else.
- b) Write a shell script that accepts one or more file name as arguments and converts all of them to uppercase, provided they exist in the current directory.

Week6

- a) Write a shell script that accepts a file name starting and ending line numbers as arguments and displays all the lines between the given line numbers.
- b) Write a shell script that deletes all lines containing a specified word in one or more files supplied as arguments to it.

Week7

- a) Write a shell script that computes the gross salary of an employee according to the following rules:
 - i) If basic salary is < 1500 then HRA =10% of the basic and DA =90% of the basic.
 - ii) If basic salary is >=1500 then HRA =Rs500 and DA=98% of the basic
 The basic salary is entered interactively through the key board.
- b) Write a shell script that accepts two integers as its arguments and computes the value of first number raised to the power of the second number.

Week8

- a) Write an interactive file-handling shell program. Let it offer the user the choice of copying, removing, renaming, or linking files. Once the user has made a choice, have the program ask the user for the necessary information, such as the file name, new name and so on.
- b) Write shell script that takes a login name as command – line argument and reports when that person logs in.
- c) Write a shell script which receives two file names as arguments. It should check whether the two file contents are same or not. If they are same then second file should be deleted.

Week9

- a) Write a shell script that displays a list of all the files in the current directory to which the user has read, write and execute permissions.
- b) Develop an interactive script that ask for a word and a file name and then tells how many times that word occurred in the file.
- c) Write a shell script to perform the following string operations:
 - i) To extract a sub-string from a given string.
 - ii) To find the length of a given string.

Week10

- a) Write a shell script to generate multiplication table.
- b) Write a shell script to print sum of individual digits of a number.

Week11

Write C programs that simulate the following unix commands:

- a) mv
- b) cp (Use system calls)

Week12

Write a C program that simulates ls Command(Use system calls / directory API)

TEXT BOOKS

1. Introduction to UNIX & SHELL programming, M.G. Venkatesh Murthy, Pearson Education.
2. Unix concepts and applications, Fourth Edition, Sumitabha Das, TMH.
3. Unix for programmers and users, 3rd edition, Gaham Glass & K. Ables, pearson education.
4. Unix and shell Programming –A text book, B.A. Forouzan & R.F. Giberg, Thomson.
5. Beginning shell scripting, E. Foster – Johnson & other, Wile Y- India.

(A0582126) DESIGN AND ANALYSIS OF ALGORITHMS LAB

1. Write a program to perform Heap sort and insertion sort for any given list of numbers.
2. Write a program to implement disjoint set operations.
3. Write a program to implement DFS and BFS of a given graph.
4. Write a program to check whether given graph is connected or not.
5. Write a program to perform Quick Sort for the given list of integer values.
6. Write a program to find Maximum and Minimum of the given set of integer values.
7. Write a Program to perform Merge Sort on the given two lists of integer values.
8. Write a program to find solution for knapsack problem using greedy method.
9. Write a program to find minimum cost spanning tree using Prim's Algorithm.
10. Write a program for all pairs shortest path problem.
11. Write a program to solve N-QUEENS problem.

(A0528127) COMPUTER GRAPHICS**OBJECTIVES:**

This course is designed to provide a comprehensive introduction to computer graphics leading to the ability to understand contemporary terminology, progress, issues, and trends.

- The interdisciplinary nature of computer graphics is emphasized in the wide variety of examples and applications.
- Course material is structured to meet the needs of both designers and users of interactive computer graphics systems
- Provide an understanding of how a computer draws the fundamental graphics primitives - lines and filled polygons in both 2-D and 3-D.
- Understand the fundamental mathematics involved in generating a 3-D scene. Includes coordinate systems, transformations, and vector operations.
- Understand the 3-D graphics pipeline, i.e. the steps taken to transform and draw an object. This includes 3-D transformations, lighting and shading, rasterization and texture mapping. Programmable shaders will be introduced.
- A thorough introduction to computer graphics techniques, focusing on 3D modelling, image synthesis, and rendering. Topics cover: geometric transformations, geometric algorithms, software systems (OpenGL), 3D object models (surface, volume and implicit), visible surface algorithms, image synthesis, shading and mapping, ray tracing, radiosity, global illumination, photon mapping, and anti-aliasing.
- Apply knowledge gained in a series of exercises using OpenGL that demonstrate the fundamental principles of computer graphics. A functional scene graph will be developed to render 3-D scenes with lighting, texture mapping, and a variety of 3-D objects

OUTCOMES:

- Be familiar with drawing primitive objects (lines, circles, polygons) on a display.
- Be exposed to graphical input and output devices
- Master two dimensional modelling and 2-D transformations.
- Be familiar with master-instance structure.
- Master three dimensional modelling and 3-D transformations.
- Be familiar with projection of 3-D objects on a 2-D plane.
- Master clipping, fill, and rendering techniques.
- Be exposed to color and shading models.
- Be familiar with the visible surface detection methods.
- Animation techniques and languages

UNIT I

A Survey of Computer Graphics: Overview of graphics systems: Video-Display Devices, Raster-scan Systems, Random-scan Systems, Graphics Monitors and Work stations, Input devices, Hardcopy Devices and Graphics Software.

UNIT II

Output Primitives: Points and lines, Line drawing algorithms- DDA, Bresenham's line algorithm, Circle generation algorithm and Ellipse Generating algorithms.

UNIT III

2-D Geometrical transforms: Basic Transformations, Matrix representations and Homogeneous coordinates, Composite transforms, Other Transformations, Transformations between coordinate systems.

UNIT IV

2-D Viewing: Definition of view port, clipping and window. The Viewing Pipeline, Viewing coordinate reference frame, Window to View-port coordinate transformation, 2D Viewing functions, Clipping Operation, Point Clipping, Line Clipping: Cohen-Sutherland and Liang - Barsky line clipping algorithms, Polygon Clipping: Sutherland – Hodgeman polygon clipping algorithm.

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**UNIT V**

3-D Geometric and Modelling Transformations: Translation, Rotation, Scaling, other Transformations, Composite Transformations.

UNIT VI

3-D viewing: Viewing Pipeline, Viewing coordinates, Projections.

Computer Animation: Design of Animation Sequence, General Computer Animation functions, Raster Animation, Computer Animation Languages, Key-Frame systems, Motion Specifications.

TEXT BOOKS:

1. "Computer Graphics C version", Donald Hearn and M. Pauline Baker, Pearson education.

REFERENCES:

1. "Computer Graphics Principles & practice", second edition in C, Foley, VanDam, Feiner and Hughes, Pearson Education.
2. "Computer Graphics Second edition", Zhigand xiang, Roy Plastock, Schaum's outlines, Tata Mc Graw hill edition.
3. "Procedural elements for Computer Graphics", David F Rogers, Tata Mc Graw hill, 2nd edition.
4. "Principles of Interactive Computer Graphics", Neuman and Sproul, TMH.
5. "Principles of Computer Graphics", Shalini, Govil-Pai, Springer.
6. "Computer Graphics", Steven Harrington, TMH

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. I- Sem (CSE)

T C
3+1* 3**(A0527127) WEB TECHNOLOGIES & PROGRAMMING**

(Common to CSE,ECE& EEE)

OBJECTIVES:

- This course demonstrates an in-depth understanding of the tools and Web technologies necessary for business application design and development. The course covers client side scripting like HTML, JavaScript and server side scripting like servlets, JSPs. And also XML and web servers and database interfacing.

OUTCOMES:

- The main learning outcomes are:
- Development of a business application.
- Implementation of given client side and server side technologies.
- Design and develop static and dynamic web pages.
- Validate web page data with database data.

UNIT I

Introduction to HTML - HTML common tags, HTML program structure, Attributes, List, Tables, images, image maps, forms, Frames; Cascading Style sheets;

UNIT II

JavaScript - Introduction to Java Scripts, Objects in Java Script, Dynamic HTML with Java Script, Data Validation using Java Script.

UNIT III

XML - Document type definition, XML Schemas, Document Object model, Presenting XML, Using XML Processors: DOM and SAX

UNIT IV

More On Servlets – Reading Initialization parameters, the javax.servlet HTTP package, Handling Http Request & Responses, Using Cookies-Session Tracking.

JSP Application Development - Generating Dynamic Content, Using Scripting Elements Implicit JSP Objects, Conditional Processing – Displaying Values Using an Expression to Set an Attribute, Declaring Variables and Methods Sharing Data Between JSP pages, Sharing Session and Application Data.

UNIT VI

Struts: Tomcat & Struts Installation, Struts Request life cycle, Struts Configuration file, Form Validation with Struts, Simple Struts application.

UNIT VI

Database Access - Database Programming using JDBC, Types of JDBC Drivers, Studying Javax.sql.* package, Accessing a Database from a JSP Page, Application – Specific Database Actions.

AJAX – Introduction, Background, How AJAX works, Common steps AJAX will follow.

TEXT BOOKS:

1. HTML Black Book – Steve Holzner.
2. Web Programming, building internet applications, Chris Bates 2nd edition, WILEY Dreamtech
3. The complete Reference Java 2 Fifth Edition by Patrick Naughton and Herbert Schildt. TMH
4. Java Server Pages – Hans Bergsten, SPD O'Reilly

REFERENCE BOOKS:

1. Programming world wide web-Sebesta, Pearson.
2. Core SERVLETS AND JAVASERVER PAGES VOLUME 1: CORE TECHNOLOGIES By Marty Hall and Larry Brown Pearson.
3. Internet and World Wide Web – How to program by Dietel and Nieto PHI/Pearson Education Asia.
4. Jakarta Struts Cookbook, Bill Siggelkow, S P D O'Reilly
5. Murach's beginning JAVA JDK 5, Murach, SPD.
6. An Introduction to web Design and Programming – Wang-Thomson.

(A0529127) DATA WAREHOUSING & MINING**OBJECTIVES:**

The main objective of this course is to provide students

- With the basic data warehousing and data mining concepts
- To learn mining rules in large databases
- To get idea on clustering analysis
- Applications that can enable them to set up and manage an industrial data warehousing and data mining system.
- To learn about OLTP and OLAP systems

OUTCOMES:

Upon completion of the course students should:

- Ability to do Conceptual, Logical, and Physical design of Data Warehouses
- Familiarity with Requirements Engineering for Data Warehouses
- OLAP applications and OLAP deployment
- Have a good knowledge of the fundamental concepts that provide the foundation of data mining.
- Learn broad classes of data mining technologies
- Understand how these concepts are engineered to use some of the basic data mining tools.

UNIT – I

Introduction: What Motivated Data Mining? Why is it Important?; What is Data Mining?; Data Mining-On What Kind of Data?; Data Mining Functionalities: What kinds of Data Can be Mined?; Are all of Patterns Interesting?; Classification of Data Mining Systems; Data Mining task primitives;

Data Warehouse and OLAP Technology: What is a Data Warehouse?; A Multidimensional Data Model: From Tables and Spreadsheet to Data Cubes, Stars, Snowflakes and Fact constellation schemas for Multidimensional Databases, Measures: Their Categorization and Computation, Concept Hierarchies, OLAP operations in the Multidimensional Data Model; Data Warehouse Architecture: Steps for the Design and Construction of Data Warehouses, A three-tier Architecture.

UNIT-II

Data Pre-processing: Why pre-process the data; Descriptive Data Summarization: Measuring the Central Tendency, Measuring the Dispersion of Data, Graphic Displays of Basic Descriptive Data Summaries; Data Cleaning: Missing values, Noisy Data Cleaning as a process; Data Integration and Transformation: Data Integration, Data Transformation, Data Reduction: Data Cube aggregation, attribute subset selection; Dimensionality Reduction, Luminosity Reduction;

UNIT-III

Mining Frequent patterns, Associations, and Correlations: Basic Concepts; Efficient and Scalable Frequent Itemset Mining methods: The Apriori Algorithm, Generating Association Rules from Frequent Itemsets, Improving Efficiency of Apriori, Mining Frequent Itemsets without Candidate Generation; Mining various kinds of Association Rules: Mining multilevel & multi-dimensional association rules; From Association Mining to Correlation Analysis: Strong Rules are not necessarily Interesting, From Association analysis to Correlation analysis;

UNIT-IV

Classification I: Overview of Classification and Prediction: What is Classification, What is prediction?; Issues Regarding Classification and Prediction: Preparing data for Classification and Prediction, Comparing Classification and Prediction Methods; Bayesian Classification: Bayes' theorem, Naïve Bayesian Classification; Classification by Decision Tree Induction: Decision Tree Induction, Attribute Selection Measures, Tree Pruning, Scalability and Decision Tree Induction; Rule-Based Classification: Using IF-THEN rules for Classification, Rule Extraction from Decision Tree, Rule Induction using a Sequential Covering Algorithm; Classification by Back propagation: A Multilayer Feed-Forward Neural Network, Defining Network Topology, Back propagation;

UNIT-V

Classification II and Prediction: Support Vector Machines: The Case when the Data are Linearly Separable, The Case when the Data are Linearly Inseparable; Lazy Learners: k-Nearest-Neighbour Classifiers, Case-Based Reasoning; Prediction: Linear Regression, Nonlinear Regression; Accuracy and Error Measures: Classifier Accuracy Measures, Predictor Error Measures; Evaluating the Accuracy of a Classifier or Predictor: Holdout Method and Random sub sampling, Cross validation, Bootstrap;

UNIT-VI

Cluster Analysis: Overview of Cluster Analysis; Types of data in Cluster Analysis: Interval-Scaled Variables, Binary Variables, Categorical, Ordinal, and Ratio-Scaled variables, Variables of Mixed Types; A Categorization of Major Clustering Methods; Partitioning Methods: Classical Partitioning Methods: k-Means and k-Medoids, Partitioning Methods in Large Databases: From k-Medoids to CLARANS; Hierarchical Methods: Agglomerative and Divisive Hierarchical Clustering, BIRCH, ROCK; Density-Based Methods: DBSCAN; Grid-Based Methods: STING; Model-Based Clustering Methods: Expectation-Maximization;

TEXT BOOKS:

1. Data Mining – Concepts and Techniques - Jiawei Han & Micheline Kamber Harcourt India, second Edition.

REFERENCES:

1. Data Mining Introductory and advanced topics–Margaret H Dunham, Pearson Education
2. Data Mining Techniques – Arun K Pujari, University Press.
3. Data Warehousing in the Real World – Sam Anahory & Dennis Murray. Pearson Edn Asia.
4. Data Warehousing Fundamentals – Paulraj Ponnaiah Wiley Student Edition
5. The Data Warehouse Life cycle Tool kit – Ralph Kimball Wiley Student Edition.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. I-Sem (CSE)

T	C
3+1*	3

(A0530127) SOFTWARE PROJECT MANAGEMENT

(Common to CSE&IT)

Course Objective:

The objective of the course is to familiarise students in practice with the initiation, management and supervision of a software project. During the course, actual software projects are defined and their implementation is managed and supervised. To provide basic project management skills with a strong emphasis on issues and problems associated with delivering successful IT projects. The module is designed to provide an understanding of the particular issues encountered in handling IT projects and to offer students methods, techniques and 'hands-on' experience in dealing with them.

LEARNING OUTCOMES:

The student will be able to:

1. Understand and practice the process of project management and its application in delivering successful IT projects;
2. evaluate a project to develop the scope of work, provide accurate cost estimates and to plan the various activities;
3. understand and use risk management analysis techniques that identify the factors that put a project at risk and to quantify the likely effect of risk on project timescales;
4. identify the resources required for a project and to produce a work plan and resource schedule;
5. monitor the progress of a project and to assess the risk of slippage, revising targets or counteract drift;
6. Distinguish between the different types of project and follow the stages needed to negotiate an appropriate contract.

UNIT I: CONVENTIONAL SOFTWARE MANAGEMENT

The waterfall model, conventional software Management performance.

Evolution of Software Economics: Software Economics, pragmatic software cost estimation.

UNIT II: IMPROVING SOFTWARE ECONOMICS

Reducing Software product size, improving software processes, improving team effectiveness, improving automation, achieving required quality, peer inspections.

The old way and the new: The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process.

UNIT III: LIFE CYCLE PHASES

Engineering and production stages, inception, Elaboration, construction, transition phases. Artefacts of the process: The artefact sets, Management artefacts, Engineering artefacts, programmatic artefacts.

UNIT IV: MODEL BASED SOFTWARE ARCHITECTURES

A Management perspective and technical perspective.

Work Flows of the process: Software process workflows, Iteration workflows.

UNIT V: CHECKPOINTS OF THE PROCESS

Major mile stones, Minor Milestones, Periodic status assessments.

Iterative Process Planning: Work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning.

UNIT VI: PROJECT ORGANIZATIONS AND RESPONSIBILITIES

Line-of-Business Organizations, Project Organizations, evolution of Organizations.

Process Automation: Automation Building blocks, The Project Environment.

TEXT BOOKS:

1. Software Project Management, Walker Royce: Pearson Education, 2005.

REFERENCE BOOKS:

1. Software Project Management, Bob Hughes and Mike Cotterell: Tata McGraw-Hill Edition.
2. Software Project Management, Joel Henry, Pearson Education.
3. Software Project Management in practice, PankajJalote, Pearson Education.2005.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. I-Sem (CSE)

T	C
3+1*	3

(A0531127) SAP-ABAP AND BASIC APPLICATIONS

(Common to CSE & IT)

(Elective II)

OBJECTIVES:

- Student will learn about the SAP ABAP Programming Language and how to use the SAP ABAP Workbench tools to develop basic applications.
- The course objective is to understand the concepts of Procedure and Object Oriented Programming in SAP ABAP to develop various applications, and to understand the concepts of ABAP Objects.
- It allows understanding the Data Dictionary and Open SQL which include in basic concepts of SAP ABAP to manipulate the data in database.

EXPECTED OUTCOMES:

- Understand and use the basic programming concepts of SAP ABAP.
- Design, develop the code, check, activate and run the programs and database tables using SAP ABAP Workbench tools.
- Understand and use the basic SAP ABAP application concepts to be able to develop the applications.
- After completing the course, participants should able to : Understand the advantage of ERP tools, working with SAP ABAP Workbench Tools, basic programming concepts in ABAP, Working with database through ABAP data dictionary, Open SQL and Internal Tables, Understand and develop the basic applications.

UNIT - I

What is ERP?, Why we need ERP?, Advantages of ERP, Major ERP Packages, What is SAP?, History & Features of SAP, SAP R/2 Architecture (Limitations of R/2 Architecture), SAP R/3 Architecture (Types of work processes), SAP R/3 Application Modules, SAP Landscape, What is ABAP?, Logon to SAP Environment, Transaction Codes.

Programming Concepts: ABAP/4 Editor (SE38), Steps for Creating a Program, Elements in R/3 Screen, ABAP Syntax, Comments, Errors, Write Statements, Data, Data types, variables, parameters, system variables, control statements, string operations.

UNIT - II

ABAP Dictionary: Introduction, Exploring Domain, Data types, Types Groups, Database Tables, structures, append structures, views, and search helps, lock object, Primary key and foreign key.

Internal Table: Introduction, types of internal table, Declaring Internal Table, Populating Internal Table, Processing Internal Table, Initializing Internal Tables, Control Break processing.

UNIT – III

OPEN SQL: Accessing Database Tables, Reading data using select statement, insert, update, modify, delete.

Modularization Techniques: Working with subroutines, Macros, Function Modules.

UNIT - IV

REPORTS: Working with classical reports, interactive reports, ALV Reports.

UNIT – V

ABAP User Dialogues: Introduction, introducing dialog programming, screen painter, menu painter, working with selection screens.

Forms in SAP: Exploring the SAP script tool, The SAP smart Forms Tool, Comparing SAP Script and smart Forms, migrating SAP script forms to smart Forms.

UNIT - VI

OOPS Concepts: Encapsulation, Abstract, Polymorphism, Inheritance, Defining OOPS Concepts local and Global.

TEXT BOOKS:

1. SAP ABAP/4, Covers SAP ECC 6.0 Black Book, Kogent Learning Solutions Inc., DreamTech Press.
2. "Introduction to ABAP/4 programming for SAP" by Gareth M.de.Bruyn & Robert Lyfareff; Publisher: Golgotia pub.

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. I-Sem (CSE)

T C
3+1* 3**(A0532127) NETWORK PROGRAMMING**

(ELECTIVE - II)

OBJECTIVES:

- Understand fundamental concepts of computer communication
- Understand sockets and ports
- To provide an opportunity to do network programming using TCP/IP.
- Create comprehensive network applications using sockets

OUTCOMES:

- On completion of this course, Student should:
- Understand the key protocols that support the Internet;
- Be familiar with several common programming interfaces for network communication;
- Have a detailed knowledge of the TCP/UDP Sockets
- Create applications using techniques such as multiplexing, forking, multithreading;
- I/O, non-blocking I/O and event driven I/O.
- Apply knowledge of Unix/Linux operating systems to build robust client and server software for this environment;
- Learn advanced programming techniques such as IPv6 Socket Programming, Broadcasting, Multicasting

UNIT-I

Introduction to Network Programming: OSI model, Unix standards, TCP and UDP & TCP connection establishment and Format, Buffer sizes and limitation, standard internet services, Protocol usage by common internet application.

UNIT-II

Sockets: Address structures, value – result arguments, Byte ordering and manipulation function and related functions Elementary TCP sockets – Socket, connect, bind, listen, accept, fork and exec function, concurrent servers. Close function and related function.

UNIT-III

TCP client server: Introduction, TCP Echo server functions, Normal start up, terminate and signal handling server process termination, Crashing and Rebooting of server host shutdown of server host.

UNIT-IV

I/O Multiplexing and socket options: I/O Models, select function, Batch input, shutdown function, poll function, TCP Echo server, getsockopt and setsockopt functions. Socket states, Generic socket option IPV6 socket option ICMPV6 socket option IPV6 socket option and TCP socket options.

UNIT-V

Elementary UDP sockets: Introduction UDP Echo server function, lost datagram, summary of UDP example, Lack of flow control with UDP, determining outgoing interface with UDP.

Elementary name and Address conversions: DNS, get host by Name function, Resolver option, Function and IPV6 support, uname function, other networking information.

UNIT-VI

IPC: Introduction, File and record locking, Pipes, FIFOs streams and messages, Name spaces, system V IPC, Message queues, Semaphores.

TEXT BOOKS:

1. UNIX Network Programming, Vol. I, Sockets API, 2nd Edition. - W.Richard Stevens, Pearson Edn. Asia.
2. UNIX Network Programming, 1st Edition, - W.Richard Stevens. PHI.

REFERENCES:

1. UNIX Systems Programming using C++ T CHAN, PHI.
2. UNIX for Programmers and Users, 3rd Edition Graham GLASS, King abls, Pearson Education
3. Advanced UNIX Programming 2nd Edition M. J. ROCHKIND, Pearson Education.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. I-Sem (CSE)

T C
3+1* 3**(A0423127) DIGITAL IMAGE PROCESSING**

(ELECTIVE II)

(Common to CSE, EIE &ECE)

OBJECTIVES:

- To learn the fundamentals of Image Processing.
- To learn sampling and reconstruction procedures.
- To learn the various transforms used in image Processing.
- To study various concepts of image enhancement, reconstruction and image compression.
- To design image processing systems.

OUTCOMES:

- Develops ability to identify, formulate & solve problems involving images.
- Develops ability to design & conduct experiments, analyze & interpret image data.
- To design a software, Component or process as per needs & specifications.
- It will demonstrate the skills to use modern engineering tools, software's & equipment to analyze problems.
- Develop confidence for self-education & ability for life-long learning.
- It will show the ability to participate & try to succeed in competitive Exams.

UNIT I

DIGITAL IMAGE FUNDAMENTALS: Introduction, Image sensing & acquisition, Concept of gray levels. Gray level to binary image conversion. Sampling and quantization. Relationship between pixels. Imaging Geometry, operations on digital image: array Vs matrix, linear Vs non-linear, arithmetic operations, set and logical operations, spatial operations, vector and matrix operations, probabilistic methods.

UNIT II

IMAGE TRANSFORMS: 2-D FFT, Properties. Walsh transform, Hadamard Transform, Discrete cosine Transform, Haar transform, wavelet transform, Hotelling transform, comparison of different image transforms.

UNIT III

IMAGE ENHANCEMENT: Enhancement in Spatial Domain: Point processing. Histogram processing. Spatial filtering (Smoothing and sharpening), Enhancement in frequency domain: Basics of filtering in frequency domain, Image smoothing, Image sharpening, Homomorphic filtering, basics of colour image processing.

UNIT IV

IMAGE RESTORATION: Noise models, Degradation model, Restoration in the presence of noise only, Spatial filtering, Inverse filtering, Least mean square filters, Constrained Least Squares Restoration, Interactive Restoration.

UNIT V

IMAGE SEGMENTATION: Introduction, Detection of discontinuities. Edge linking and boundary detection, Thresholding, Region based segmentation. Use of motion in segmentation.

UNIT VI

IMAGE COMPRESSION: Need for image compression, Redundancies and their removal methods, Fidelity criteria, Image compression models, Source encoder and decoder, Error free compression, Lossy compression.

TEXT BOOK :

1. Digital Image processing – R.C. Gonzalez & R.E. Woods, Addison Wesley/ Pearson education, 2nd Edition, 2002.
2. Digital image processing by S.Jayaraman, S.Esakkirajan & T.Veera Kumar, Tata McGraw Hill, 2010.

REFERENCES :

1. Fundamentals of Digital Image processing – A.K.Jain, PHI.
2. Digital Image processing using MAT LAB – Rafael C. Gonzalez, Richard E Woods and Steven L. Edition, PEA, 2004.
3. Digital Image Processing – William K. Pratt, John Wiley, 3rd Edition, 2004.
4. Fundamentals of Electronic Image Processing – Weeks Jr., SPIC/IEEE Series, PHI.

**(A1220126) EMBEDDED COMPUTING
(ELECTIVE-II)****Course Objectives:**

- To distinguish the characteristics of embedded computer systems.
- To examine the various vulnerabilities of embedded computer systems.
- To evaluate/critique various protection methodologies as to their effectiveness to deter, detect, and respond to exploitation activities.

Learning Outcomes:

The student will be able to:

- Acquire a working knowledge of embedded computer hardware and software architecture.
- Use Hex Editors and Basic Debuggers to navigate, reverse engineer, and modify machine-readable code.
- Examine the operation, implementation, and circumvention of various commonly employed data integrity algorithms.

UNIT I

Embedded Computing: Complex Systems and Microprocessors, The Embedded System Design Process, Formalisms for System Design, Model Train Controller.

Instruction Sets: ARM Processor.[1.1 - 1.4 and 2.2]

UNIT II

CPUs: Programming Input and Output, Supervisor Mode, Exceptions, and Traps, CPU Performance, CPU Power Consumption, Design Example: Data Compressor.

Bus-Based Computer Systems: The CPU Bus, I/O devices, Component Interfacing, Designing with Microprocessors, Development and Debugging, System-Level Performance Analysis, Design Example: Alarm Clock.[3.1,3.2,3.5 ,3.6, 3.7 and 4.1,4.3,4.4,4.5,4.6,4.7,4.8]

UNIT III

Program Design and Analysis: Components for Embedded Programs, Models of Programs, Program Optimization, Program-Level Performance Analysis, Software Performance Optimization, Program-Level Energy and Power Analysis and Optimization, Analysis and Optimization of Program Size, Software Modem.[5.1, 5.2, 5.5, 5.6, 5.7, 5.8, 5.9, 5.11]

UNIT IV

Processes and Operating Systems: Multiple Tasks and Multiple Processes, Preemptive Real-Time Operating Systems, Priority-Based Scheduling, Interprocess Communication Mechanisms, Evaluating Operating System Performance, Power Management and Optimization for Processes, Design Example: Telephone Answering Machine.[6.1,6.2, 6.3, 6.4, 6.5, 6.6, 6.7]

UNIT V

Multiprocessors: Why Multiprocessors?, CPU's and Accelerators, Multiprocessor Performance Analysis, Consumer Electronics Architecture, Design Examples: Cell Phones, Compact DISCs and DVDs, Audio Players, Digital Still Cameras, Video Accelerator.[7.1, 7.2, 7.3, 7.4, 7.5, 7.6, 7.7, 7.8, 7.9]

UNIT VI

Networks: Distributed Embedded Architectures, Networks for Embedded Systems, Network-Based Design, Internet-Enabled Systems, Vehicles as Networks, Sensor Networks, Design Example: Elevator Controller. [8.1, 8.2, 8.3, 8.4, 8.5, 8.6, 8.7]

TEXT BOOK:

1. Computers as Components - Principles of Embedded Computing System Design, by Wayne Wolf, 2/E, Morgan Kaufmann Publishers, 2012.

REFERENCES BOOKS:

1. Embedded Realtime Systems Programming, Sriram V. iyer, Pankaj Gupta, Tata McGraw-Hill Pub. Co. Ltd, 2004. [Chaps 2, 3.2, 4-7]
2. Embedded Systems, Raj Kamal, -Hill Pub. Co. Ltd, 11th print 2007. [Chaps 1-5, Appendix G]
3. An Embedded Software Primer, David E. Simon, Pearson Education, 2007. [Chps 5-10]
4. Programming for Embedded Systems, Dream Software Team, WILEY dreamtech India Ltd. 2005. [has lots of excellent case studies]
5. Embedding system building blocks, Labrosse, via CMP publishers.

**(A0533127) DISTRIBUTED DATABASES
(ELECTIVE - III)****OBJECTIVES:**

The first is study of the classical distributed database management

- Issues in Distribution design, distributed query processing, and distributed transaction management.
- To learn about Optimization in DDB
- Architectural Issues in DDB

OUTCOMES:

Upon completion of the course students are able to

- Understand distributed database management;
- Good knowledge of complex topics like Distribution design, distributed query processing
- Solution to the some of the Architectural issues.
- Examples of DDB's

UNIT I

Introduction to Distributed Databases: Features of Distributed versus Centralized Databases, why distributed databases?, Distributed Database Management Systems, Review of databases, Review of computer networks.

UNIT II

Levels Of Distribution Transparency: Reference Architecture for Distributed Databases , Types of Data Fragmentation, Distribution transparency for read only applications, Distribution transparency for update applications, distributed database access primitives, Integrity Constraints in Distributed Databases.

UNIT III

Distributed Database Design: A Framework for Distributed database Design, The Design of database Fragmentation

UNIT IV

Translation of Global Queries to Fragment Queries: Equivalence Transformations for Queries, Transforming Global Queries into Fragment Queries, Distributed Grouping and aggregate Function Evaluation, Parametric Queries.

UNIT V

The Management of Distributed Transactions: A Framework for Transaction Management Supporting Atomicity of Distributed Transactions, Concurrency Control for Distributed Transactions, Architectural Aspects of Distributed Transactions.

UNIT VI

Concurrency Control: Foundation of Distributed Concurrency Control, Distributed Deadlocks, Concurrency Control based on Timestamps, Optimistic Methods for Distributed Concurrency Control.

TEXT BOOKS :

1. Distributed Database Principles & Systems, Stefano Ceri, Giuseppe Pelagatti McGraw-Hill

REFERENCES:

1. Principles of Distributed Database Systems, M.Tamer Ozsu, Patrick Valduriez – Pearson Education.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. I-Sem (CSE)

T C

3+1* 3

(A0534127) FUNDAMENTALS OF SOFT COMPUTING

(ELECTIVE - III)

Course Objectives:

- Soft computing covers non-traditional techniques or approaches for solving real-world problems. Content of course, in accordance with meaning of its name, is as follow: Tolerance of imprecision and uncertainty as the main attributes of soft computing theories. Neural networks, Fuzzy Logic, Genetic Algorithms.

Learning Outcomes:

- The Student will be able to acquire knowledge of soft computing theories fundamentals and so they will be able to solve problems using approaches of these theories for solving various real-world problems.

UNIT I

Introduction: Hard Computing: Features of Hard Computing, Soft Computing: Features of Soft Computing, Hybrid Computing.

Optimization and Some Traditional Methods: Introduction to optimization: A Practical Example, Classification of Optimization Problems, Principle of Optimization, Duality Principle, Traditional Methods of Optimization: Exhaustive Search Method, Random Walk Method, Steepest Descent Method, Drawbacks of Traditional Optimization Methods.

UNIT II

Introduction to Genetic Algorithms: Working Cycle of a Genetic Algorithm, Binary-Coded GA: Crossover or Mutation?, A Hand Calculation, Fundamental theorem of GA/Schema Theorem, Limitations of a Binary-Coded GA, GA-parameters Setting, Constraints Handling in GA: Penalty Function Approach, Advantages and Disadvantages of Genetic Algorithm.

UNIT III

Some Specialized Genetic Algorithms: Real-Coded GA: Crossover Operators, Mutation Operators, Micro-GA, Visualized Interactive GA: Mapping Methods, Simulation Results, Working Principle of the VIGA. Scheduling GA: Edge Recombination, Order Crossover #1, Order Crossover #2, Cycle Crossover, Position-Based Crossover, Partially Mapped Crossover(PMX)

UNIT IV

Introduction to Fuzzy Sets: Crisp Sets: Notations Used in Set Theory, Crisp Set Operations, Properties of Crisp Sets, Fuzzy Sets: Representation of Fuzzy Set, Difference Between Crisp Set and Fuzzy Set, A Few Definitions in Fuzzy Sets, Properties of Fuzzy Sets.

UNIT V

Fuzzy Reasoning and Clustering: Introduction, Fuzzy Logic Controller: Two Major Forms of Fuzzy Logic Controller, Hierarchical Fuzzy Logic Controller, Sensitivity Analysis, Advantages and Disadvantages of Fuzzy Logic Controller, Fuzzy Clustering: Fuzzy C-Means Clustering, Entropy-based Fuzzy Clustering.

UNIT VI

Fundamentals of Neural Networks: Introduction: Biological Neuron, Artificial Neuron, A Layer of Neurons, Multiple Layers of Neurons, Static vs. Dynamic Neural Networks, Training of Neural Networks: Supervised Learning, Un-supervised Learning, Batch Training.

TEXT BOOK:

1. "Soft Computing", D.K.Pratihar, Narosa Publishing House.

REFERENCE BOOK:

1. Neuro-Fuzzy and Soft Computing, J.S.R Jan, C.T Sun and E.Mizutani, PHI,2005.
2. "Soft Computing and Intelligent Systems Design", Fakhreddine O. Karray and Clarence De Silva, Pearson Education,2009.

(A0535127) OBJECT ORIENTED SOFTWARE ENGINEERING
(ELECTIVE-III)**OBJECTIVES:**

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

- Describe and explain concepts and principles of object oriented software development
- Describe and explain fundamental theories, techniques and methods in software engineering
- Master basic object oriented modelling principles
- Describe and explain basic concepts and constructs in the Java programming language
- Implement programs in the Java programming language
- Individually, or in teams, perform specific software development task; collecting and analyzing requirements; develop and evaluate a design; test and document an implementation
- Individually, or in teams, perform a software development project using object oriented technology

OUTCOMES:

- Analyze and model requirements and develop software using object-oriented analysis and design.
- Express object models in UML
- Use CASE tools for software design.
- Work as a member of a software development team.

UNIT I

Introduction to Classical software Engineering: Historical, Economic and Maintenance aspects. Introduction to OO Paradigm. Different phases in structured paradigm and OO Paradigm. Software Process and different life cycle models and corresponding strengths and weaknesses.

UNIT II

Planning and Estimation: Estimation of Duration and Cost , COCOMO components of software. Project Management plan

UNIT III

Requirements phase: Rapid Prototyping method, Formal methods of developing specification document, Examples of other semi - formal methods of using Finite-State- Machines, Petri nets.

UNIT IV

Analysis phase: Use case Modeling, Class Modeling , Dynamic Modeling, Testing during OO Analysis

UNIT V

Design phase: Data oriented design, Object Oriented design, Formal techniques for detailed design.

UNIT VI

IIM Phases: Implementation , Integration and maintenance phases, OOSE aspects in these phases

TEXT BOOKS

1. Object oriented and Classical Software Engineering, 7/e, Stephen R. Schach, TMH
2. Object oriented and classical software Engineering, Timothy Lethbridge, Robert Laganieri, TMH

REFERENCE BOOKS

1. Component-based software engineering: 7th international symposium, CBSE 2004, Ivica Crnkovic, Springer

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. I-Sem (CSE)

T C

3+1* 3

**(A0536127) ENTERPRISE APPLICATION INTEGRATION
(ELECTIVE-III)****Objectives**

- The course aims at introducing and discussing in depth several important topics related to distributed information systems in general and enterprise application integration in particular. In many ways, the course explores the synergy between information and communication systems and how this synergy can be best exploited for EAI and B2B integration.

Outcomes

- Learn how to develop and reuse software components in web applications.
- Work in groups to efficiently develop and deploy a web application.
- Understand the enterprise application integration.

UNIT I

Approaching Application Integration: Moving from Information Oriented to service oriented Application Integration, Application Integration Approaches(CH1).

UNIT –II

Types Of Application Integration: Information oriented Application Integration, Business Process Integration oriented Application Integration (Ch2 and CH3).

UNIT –III

Types of Application Integration: Service Oriented Application Integration The Basics, Enter Web Services, Web services Exposed, Scenarios, Understanding Service Frame works, Moving To application Services. (CH4).

UNIT-IV

Portal Oriented Application Integration: POAI Example, Portal Power, Web enabled World, Portal Architecture (Ch5).

UNIT V

Application Integration Technology: Middle Ware Basics (CH6).

UNIT VI

Middle ware types and Application Integration (Ch7).

TEXT BOOK:

1. Next Generation Application Integration: From Simple Information to Web services By David S Linthicum Addison Wesley Professional

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. I-Sem (CSE)

T	C
3	2

(A0537127) CCNA

(Skill Development Course)

COURSE OBJECTIVES:

1. The main objectives of this course are how networks function, identifying major components, function of network components and the Open System Interconnection (OSI) reference model
2. Host-to-host packet delivery process, describe issues related to increasing traffic on an Ethernet LAN and identify switched LAN technology solutions to Ethernet networking issues.
3. Describes the reasons for extending the reach of a LAN and the methods that can be used with a focus on RF wireless access.
4. Describes the reasons for connecting networks with routers and how routed networks transmit data through networks using TCP/IP
5. Use the command-line interface to discover neighbours on the network and managing the router's start up and configuration
6. Expand a small-sized, switched LAN to a medium-sized LAN with multiple switches, supporting VLANs, trunking, and spanning tree
7. Configure, verify, and troubleshoot OSPF, Configure, verify, and troubleshoot EIGRP

COURSE OUTCOMES:

1. By this course students will be able to understand the network concepts clearly.
2. By this course students will get a clear idea to install, configure, operate, and troubleshoot medium-size route and switched networks, including implementation and verification of connections to remote sites in a WAN.

Prerequisites:

The knowledge and skills that you must have before attending this course are as follows:

1. Basic computer literacy
2. Windows navigation skills
3. Basic Internet usage skills
4. Fundamental understanding of data networking and IP addressing
5. Familiarity with the Cisco IOS command-line interface

UNIT I: EXAMINING THE NETWORK

Scope of Networks, Basic Connectivity Components, Network Topologies, Network Technologies, Expanding the Network

UNIT II: EXAMINING NETWORK PROTOCOLS

Introduction to Protocols, Protocols and Data Transmissions, Common Protocols, Other Communication Protocols, Remote Access Protocols

UNIT III: EXAMINING IP ADDRESSING

Classful IP Addressing (IP Addresses, IP Address Classes),

Subnetting a Network (Subnets, Subnet Masks, Determining Local and Remote Hosts),

Planning IP Addressing (Addressing Guidelines, Assigning Network IDs, Assigning Host IDs),

Assigning TCP/IP Addresses (Static IP Addressing, Automatic IP Addressing, Viewing TCP/IP Configuration, Viewing TCP/IP Configuration Using Ipconfig)

UNIT IV: ROUTERS AND ROUTING BASICS

Routers, Routing Basics, IP Routing Table and Directly Connected Networks, Static Routing, Routing Theory and Dynamic Routing Operations.

UNIT V: ROUTING PROTOCOLS-RIP, RIPv2, OSPF, EIGRP, and IGRP**UNIT VI: SWITCHING - Types of VLANs, VTP, STP****TEXT BOOK:**

1. CCNA Study Guide v2.52, Aaron Balchunas, 2012.

REFERENCES:

1. Cisco Certified Network Associate Study Guide by Todd Lammle, 5th edition.
2. Authorized Self-Study Guide Interconnecting Cisco Network Devices, Part 1(ICND1) by Steve McQuerry, Second Edition.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. I-Sem (CSE)

P	C
3	2

(A0583127) SAP ABAP APPLICATIONS LAB

(Elective II Lab)

(Common to CSE & IT)

WEEK-1

1. Write a program to print semester details by using write keyword.
2. Write a program to display the list of modules in SAP (SD, MM, HR, PP, FI/CO, CRM, SEM).
3. Write a program to print the values in system variables.
4. Write a program to print the list of importance days in a year.

WEEK-2

1. Accept two integer values from selection screen and perform the following.
 - (a) Addition
 - (b) Subtraction
 - (c) Multiplication
 - (d) Division
 - (e) Modulo-Division.
2. Write a program to accept a number from user and print multiplication table of it.
3. Write a program to print week days by using selection screen
4. (1) IF-ENDIF
(2) IF-ELSE-IF
(3) NESTED IF

WEEK-3

1. Write a program to print year of months by using CASE-ENDCASE statement.
2. Write a program to print even numbers by using DO and ENDDO statement.
3. Write a program to print sequence of numbers by using WHILE-END WHILE statement.
4. Write a character string program by using predefined operations.
5. (a) TRANSLATE
(b) CONDENSE
(c) CONCATENATE
(d) SPLIT
(e) REPLACE
(f) STRLEN

WEEK-4

1. Create database table using the following fields by creating data elements and domains: Vendor number, name of the account holder, bank key, bank account number, bank city and country key.
2. Create the structure of some common fields: name, mobile no, street, city, country.
3. Create Employee table define domain EMPNO and remaining fields make use of INCLUDE structure.

WEEK-5

1. Create the database tables as following and make use of INCLUDE structure for common fields:
 - a) Vendor Details: Vendor No (PK), Name, City, District, Street, Country
 - b) Customer Details: Customer No (PK), Vendor No (FK), City, District, Street, Country
 - c) Company Details: Company Code (PK), Name, Customer No (FK), City, District, Street, Country.

WEEK-6

1. Write a program to find factorial of integer value using subroutine, macro and function module.
2. Create a internal table (ITAB) for KNA1 with five fields ,KUNNR, NAME1, ADRNR, ORTO1, LAND1. Display the data in ITAB and download the content to a file.

WEEK-7

1. Select Options
2. Add records to internal table ITAB
3. Delete and Adjacent duplicates
4. Sort
5. Read
6. Modify

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**WEEK-8**

1. Write a program to display the changing of database operation (Insert, Update, Delete).

WEEK-9

1. Create a report for MARA table with Four fields, MANTR, MTART, MATKL, MEINS. Display the data in report and download the content to a file.
2. Inner Join
3. Loop AT
4. AT FIRST
5. AT NEW
6. AT END
7. AT LAST

WEEK-10

1. Write a program to create interactive report.
A) ATLINE-SELECTION B) GET CURSOR C) HIDE D) BUTTON SELECTION.

WEEK-11

1. Write a program to create ALV reports by using events.

WEEK-12

1. Develop a print program.

WEEK-13

1. Create local and global classes and write programs to show inheritance.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. I-Sem (CSE)

P	C
3	2

(A0584127) NETWORK PROGRAMMING LAB

(Elective II Lab)

WEEK1.

Implement the following forms of IPC.

- a) Pipes
- b) FIFO

WEEK2.

Implement file transfer using Message Queue form of IPC

WEEK3.

Write a program to create an integer variable using shared memory concept and increment the variable simultaneously by two processes. Use semaphores to avoid race conditions

WEEK4.

Design TCP iterative Client and server application to reverse the given input sentence

WEEK5.

Design TCP iterative Client and server application to reverse the given input sentence

WEEK6.

Design TCP client and server application to transfer file

WEEK7.

Design a TCP concurrent server to convert a given text into upper case using multiplexing system call "select"

WEEK8.

Design a TCP concurrent server to echo given set of sentences using poll functions

WEEK9.

Design UDP Client and server application to reverse the given input sentence

WEEK10

Design UDP Client server to transfer a file

WEEK11

Design using poll client server application to multiplex TCP and UDP requests for converting a given text into upper case.

WEEK12

Design a RPC application to add and subtract a given pair of integers

REFERENCE BOOK:

1. Advance UNIX Programming Richard Stevens, Second Edition Pearson Education
2. Advance UNIX Programming, N.B. Venkateswarlu, BS Publication.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. I-Sem (CSE)

P	C
3	2

(A0585127) DIGITAL IMAGE PROCESSING LAB
(Elective Lab 2)

1. Write program to read and display digital image using MATLAB or SCILAB
2. Write a program to perform convolution operation for 1D and 2D data in MATLAB/SCILAB
3. To write and execute programs for image arithmetic operations
4. To write and execute programs for image logical operations
5. To write and execute program for geometric transformation of image
6. To understand various image noise models and to write programs for image restoration
7. Write and execute programs to remove noise using spatial filters
8. Write and execute programs for image frequency domain filtering
9. Write a program in C and MATLAB/SCILAB for edge detection using quick mask
10. Write a program in C and MATLAB/SCILAB for histogram calculation and equalization. (Without using standard functions)

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. I-Sem (CSE)

P	C
3	2

(A1298126) EMBEDDED COMPUTING LAB**Elective Lab 2**

1. Write a Program to
 - a) Read inputs from switches.
 - b) To make LEDs blink.
2. Write a Program for serial communication.
3. Write a Program for encryption / decryption.
4. Develop necessary interfacing circuit to read data from a sensor and process using the 801 and 8051 boards. The data to be displayed on a PC monitor.
5. Sort RTOs (mCOS) on to 89CS1 board and Verify.
6. Simulate on elevator movement using RTO's on 89CSI board.

(A0586127) WEB TECHNOLOGIES&PROGRAMMING LAB**Objective :**

- To create a fully functional website with mvc architecture. To Develop an online Book store using we can sell books (Ex amazon .com).

Hardware and Software required:

- A working computer system with either Windows or Linux
- A web browser either IE or firefox
- Tomcat web server and Apache web server
- XML editor like Altova Xml-spy [www.Altova.com/XMLSpy – free] , Stylusstudio , etc.,
- A database either Mysql or Oracle
- JVM(Java virtual machine) must be installed on your system
- BDK(Bean development kit) must be also be installed

Week-1:

Design the following static web pages required for an online book store web site.

1) HOME PAGE:

The static home page must contain three **frames**.

Top frame : Logo and the college name and links to Home page, Login page, Registration page, Catalogue page and Cart page (the description of these pages will be given below).

Left frame : At least four links for navigation, which will display the catalogue of respective links.

For e.g.: When you click the link “CSE” the catalogue for **CSE** Books should be displayed in the Right frame.

Right frame: The *pages to the links in the left frame must be loaded here*. Initially this page contains description of the web site.

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE ECE EEE CIVIL	Description of the Web Site			

Fig 1.1

2) LOGIN PAGE:

This page looks like below:

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE ECE EEE CIVIL	Login : <input type="text"/> Password: <input type="password"/> <input type="button" value="Submit"/> <input type="button" value="Reset"/>			

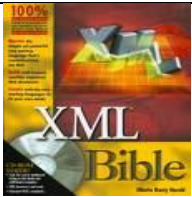

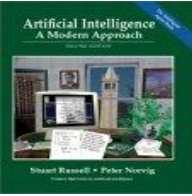





3) CATOLOGUE PAGE:

The catalogue page should contain the details of all the books available in the web site in a table.

The details should contain the following:

- Snap shot of Cover Page.
- Author Name.
- Publisher.
- Price.
- Add to cart button.

AUTONOMOUS
SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE		Book : XML Bible Author : Winston Publication : Wiely	\$ 40.5	
ECE		Book : AI Author : S.Russel Publication : Princeton hall	\$ 63	
EEE		Book : Java 2 Author : Watson Publication : BPB publications	\$ 35.5	
CIVIL		: HTML in 24 hours Author : Sam Peter Publication : Sam publication	\$ 50	

Note: Week 2 contains the remaining pages and their description.

Week-2:**4) CART PAGE:**

The cart page contains the details about the books which are added to the cart.

The cart page should look like this:

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE	Book name	Price	Quantity	Amount
ECE	Java 2	\$35.5	2	\$70
EEE	XML bible	\$40.5	1	\$40.5
CIVIL	Total amount - \$130.5			

5) REGISTRATION PAGE:

Create a "registration form" with the following fields

- 1) Name (Text field)
- 2) Password (password field)
- 3) E-mail id (text field)
- 4) Phone number (text field)
- 5) Sex (radio button)
- 6) Date of birth (3 select boxes)
- 7) Languages known (check boxes – English, Telugu, Hindi, Tamil)
- 8) Address (text area)

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**WEEK 3:****VALIDATION:**

Write *JavaScript* to validate the following fields of the above registration page.

1. Name (Name should contains alphabets and the length should not be less than 6 characters).
2. Password (Password should not be less than 6 characters length).
3. E-mail id (should not contain any invalid and must follow the standard pattern name@domain.com)
4. Phone number (Phone number should contain 10 digits only).

Note : You can also validate the login page with these parameters.

Week-4:

Design a web page using **CSS (Cascading Style Sheets)** which includes the following:

- 1) Use different font, styles:

In the style definition you define how each selector should work (font, color etc.). Then, in the body of your pages, you refer to these selectors to activate the styles.

For example:

```
<HTML>
<HEAD>
<style type="text/css">
B.headline { color:red; font-size:22px; font-family:arial; text-
decoration:underline }
</style>
</HEAD>
<BODY>
<b>This is normal bold</b><br>
Selector { cursor:value }
For example:
<html>
<head>
<style type="text/css">
.xlink { cursor:crosshair }
.hlink { cursor:help }
</style>
</head>
<body>
<b>
<a href="mypage.htm" class="xlink">CROSS LINK</a>
<br>
<a href="mypage.htm" class="hlink">HELP LINK</a>
</b>
</body>
</html>
<b class="headline">This is headline style bold</b>
</BODY>
</HTML>
```

- 2) Set a background image for both the page and single elements on the page.
You can define the background image for the page like this:

```
BODY { background-image:url (myimage.gif);}
```

- 3) Control the repetition of the image with the background-repeat property.

As background-repeat: repeat

Tiles the image until the entire page is filled, just like an ordinary background image in plain HTML.

- 4) Define styles for links as

```
A:link
A:visited
A:active
A:hover
```

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Example:

```
<style type="text/css">
A:link {text-decoration: none}
A:visited {text-decoration: none}
A:active {text-decoration: none}
A:hover {text-decoration: underline; color: red;}
</style>
```

Week-5:

Write an XML file which will display the Book information which includes the following:

- 1) Title of the book
- 2) Author Name
- 3) ISBN number
- 4) Publisher name
- 5) Edition
- 6) Price

Write a Document Type Definition (DTD) to validate the above XML file.

Display the XML file as follows.

The contents should be displayed in a table. The header of the table should be in color GREY. And the Author names column should be displayed in one color and should be capitalized and in bold. Use your own colors for remaining columns.

Use XML schemas XSL and CSS for the above purpose.

Week-6:

- 1) Install TOMCAT web server and APACHE.

While installation assign port number 4040 to TOMCAT and 8080 to APACHE. Make sure that these ports are available i.e., no other process is using this port.

- 2) Access the above developed static web pages for books web site, using these servers by putting the web pages developed in week-1 and week-2 in the document root.

Access the pages by using the urls : <http://localhost:4040/rama/books.html> (for tomcat)

<http://localhost:8080/books.html> (for Apache)

Week-7:

User Authentication: Write a Servlet which does the following job: Insert the details of the 3 or 4 users who register with the web site (week9) by using registration form. Authenticate the user when he submits the login form using the user name and password from the database

Week-8:

Install a database(Mysql or Oracle).

Create a table which should contain at least the following fields: name, password, email-id, phone number(these should hold the data from the registration form).

Practice 'JDBC' connectivity.

Write a java program/servlet/JSP to connect to that database and extract data from the tables and display them.

Experiment with various SQL queries.

Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration page (week2).

Week-9:

Write a JSP which does the following job: Insert the details of the 3 or 4 users who register with the web site (week9) by using registration form. Authenticate the user when he submits the login form using the user name and password from the database

Week-10:

Assume four users user1,user2, user3 and user4 having the passwords, pwd2,pwd3 and pwd4 respectively. Write a servlet for doing the following.

Create a Cookie and add these four user id's and passwords to this Cookie

Week-11:

Create and Run struts application and validate it using struts components.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. II-Sem (CSE)

T C
3+1* 3**(A0014125) MANAGEMENT SCIENCE**

(Common to ECE, CSE, EEE, EIE, IT & CE)

OUTCOMES:

- Students will able to know how to design the plant layout and location.
- Students will able to know importance of human resource department in organization.
- Students will able to know how the SWOT analysis helps to generate alternative corporate strategies.
- Students will able by using the contemporary practices how to survive in competitive global market.

OUTCOMES:

- Students will able to know how to design the plant layout and location.
- Students will able to know importance of human resource department in organization.
- Students will able to know how the SWOT analysis helps to generate alternative corporate strategies.
- Students will able by using the contemporary practices how to survive in competitive global market

UNIT-I

INTRODUCTION TO MANAGEMENT: Concepts of Management – Nature, Importance and Functions of Management, Taylor’s Scientific Management Theory, Fayol’s Principles of Management, Mayo’s Hawthorne Experiment, Maslow’s Theory of Human Needs, Douglas McGregor’s Theory X and Theory Y, Herzberg’s Two-Factor Theory of Motivation

UNIT-II

BASIC ISSUES IN ORGANIZATION: Designing Organic Structures of Organization (Line organization, Line and staff organization, Functional organization, Committee organization, Matrix organization, Virtual organization, Cellular organization, Team structure, Boundary less organization and Departmentation, Leadership Styles, Social responsibilities of Management

UNIT-III

OPERATIONS MANAGEMENT: Principles and Types of Plant Layout-Methods of production (Job, batch and Mass Production), Materials Management: Objectives, Need for Inventory control, EOQ, ABC Analysis, Purchase Procedure, Stores Management and Stores Records - Supply Chain Management, Marketing: Functions of Marketing, Marketing Mix, Marketing Strategies based on ProductLife Cycle., Channels of distribution.

UNIT-IV

HUMAN RESOURCES MANAGEMENT: Concepts of HRM, HRD and Personnel Management and Industrial Relations (PMIR), HRM vs. PMIR, Basic functions of HR Manager: Manpower planning, Recruitment, Selection, Training and Development, Placement, Wage and Salary Administration, Promotion, Transfer, Separation, Performance Appraisal, Grievance Handling and Welfare Administration, Job Evaluation and Merit Rating.

UNIT-V

PROJECT MANAGEMENT (PERT/CPM): Network Analysis, Programme Evaluation and Review Technique (PERT), Critical Path Method (CPM), identifying critical path, Probability of Completing the project within given time, Project Cost Analysis, Project Crashing (simple problems).

UNIT-VI

WOMEN ENTREPRENEURSHIP: Scope of Entrepreneurship among women- Promotional efforts supporting Women Entrepreneurs in India – Opportunities for women entrepreneurs – Challenges/Problems of Women Entrepreneurs – Successful cases of Women Entrepreneurs.

TEXT BOOK:

1. Aryasri: Management Science, TMH, New Delhi.

REFERENCE BOOKS:

1. Kotler Philip & Keller Kevin Lane: Marketing Management 12/e, PHI, 2007
2. Koontz & Weihrich: Essentials of Management, 6/e, TMH, 2007
3. Thomas N.Duening & John M.Ivancevich Management—Principles and Guidelines, Biztantra,2007.
4. Kanishka Bedi, Production and Operations Management, Oxford University Press, 2007.
5. Memoria & S.V.Ganker, Personnel Management, Himalaya, 25/e, 2007
6. Schermerhorn: Management, Wiley, 2007.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. II-Sem (CSE)

T C
3+1* 3**(A1227126) HUMAN COMPUTER INTERACTION**

(OPEN ELECTIVE IV)

(Common to CSE & IT)

COURSE OBJECTIVES:

1. To expose students to the central concepts of Human-Computer Interaction.
2. Establish target users, functional requirements, and interface requirements for a given computer application
3. Describe and explain user interface design principles, and apply them to designing an interface.
4. Develop quick-and-dirty interface designs using rapid prototyping methods.
5. Evaluate user interface designs through usability inspection and user models
6. Develop user studies and analyze study data to gain information about users, tasks, and interface designs.
7. Choose appropriate HCI methods for specified goals and justify the choice

LEARNING OUTCOMES:

The Student will be able to:

1. Apply HCI principles and a user-centered approach to interaction design.
2. Analyze user needs and requirements.
3. Design and develop prototypes based on user assessments (needs and requirements), while applying HCI principles and models.
4. Apply evaluation and usability testing methods to interactive products to validate design decisions.
5. Develop pre-design and post-design usability testing techniques on the developed website.
6. Assess user needs and requirements.
7. Categorize, design and develop information in proper architectural structures.
8. Create interface design prototypes based on a range of design principles and user data, and user assessments.
9. Apply prototype principles and a user-centered approach to interaction design.

UNIT I:

Introduction: Importance of user Interface – definition, Importance of good design. Benefits of good design. A brief history of Screen design.

UNIT II:

The graphical user interface – popularity of graphics, the concept of direct manipulation, graphical system, Characteristics, Web user – Interface popularity, characteristics- Principles of user interface.

UNIT III:

Design process – Understanding Human interaction with computers, Importance of human characteristics in design, Human consideration in Design, Human interaction speeds.

UNIT IV:

Screen Designing: Human Considerations in Screen Design – Interface Design Goals, Screen Meaning and purpose, organizing screen elements clearly and Meaningfully, Ordering of screen data and content – Screen navigation and flow – Visually pleasing composition – Amount of information – Focus and Emphasis – Presenting information simply and meaningfully – statistical graphics – Technological consideration in interface design.

UNIT V:

Windows: Select the Proper Kinds of Windows: Window Characteristics, Components of a Window, Window Presentation Styles, Types of Windows, Window Management, Window Operations. Select the Proper Device-Based Controls: Characteristics of Device-Based Controls, Selecting the Proper Device Based Controls.

UNIT VI:

Components: Choose the Proper Screen-Based Controls – Operable Controls, Text Entry, Selection Controls, Combination Entry/selection controls, Other Operable Controls, Custom Controls, and Presentation Controls. Write Clear Text and Messages – Words, Sentences, Messages and Text, Text for Web Pages. Icons, Multimedia, Color-What Is It? Color Uses, possible problems with colors, choosing colors.

TEXT BOOKS:

1. The essential guide to user interface design, Wilbert O Galitz, 3rd Edition, Wiley India, 2007.

REFERENCES:

1. Human – Computer Interaction. ALAN DIX, JANET FINCAY, GRE GORYD, ABOWD, RUSSELL BEALG, PEARSON.
2. Interaction Design PRECE, ROGERS, SHARPS. Wiley Dreamtech,
3. User Interface Design, Soren Lauesen , Pearson Education.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. II-Sem (CSE)

T C
3+1* 3**(A0538128) INTELLECTUAL PROPERTY RIGHTS & CYBER LAW**
(OPEN ELECTIVE IV)**OBJECTIVES**

- The course is designed with an objective of providing students an understanding of Intellectual Property Rights and Cyber laws.

OUTCOMES:

- The students should be able to describe the need for cyber law

UNIT 1:**Intellectual Property Rights-Concepts and Evolution**

Introduction to Intellectual Property Rights, Evolution of Intellectual Property Laws

Standards and Concepts in Intellectual Property, Conventions and Treaties Relating to Global Administration of Intellectual Property Rights, Protection and Classification Regional Conventions and Treaties, Organization, Jurisdiction enforcement and Administration of IPRs, IPRs and Information Technology IPRs and Bio-technology, IPRs and Traditional Knowledge, Management of Intellectual Property Rights, Law of Intellectual Property and Ethical Issues, Knowledge Driven Economy and IPR, Intellectual Property Rights in India and abroad.

UNIT -2:**Law of Patents, Copyright and Trademarks**

Introduction ,Evolution of patent Law, Scope and Purpose, Classification of Patents, Patent Law in India: Patent Act of 1970, The Patents (Amendments) Act, 2002, Patent Office and Authorities, Grant of Patent, Right and Obligation of a Patentee, Infringement of Patents, Offences and penalties, Patents and other commercial Law, Patents – International Law, Patents Law- Emerging Trends, Social Implication of Patents.

Introduction to Copyrights as forms of Intellectual Property, Copyright Law in India (Copyright Act of 1957) - meaning, Form of Copyright and Ownership Assignment/License, Registration and terms of Copyright, Copyright infringement , Offences, Remedies and Enforcement, Broad casting Organization and performers, Copyright – International Law, Introduction to trademarks, Trademarks – forms of Intellectual Property, Law of trade Marks in India (trademark act of 1999)-meaning, registration and Authorities, Right conferred by Registration and use of Trademarks, Infringement of Trademarks and passing off, Offences, remedies and enforcement, Trademarks –International Law

UNIT 3 :**Law of Designs, geographical Indications and other Intellectual Property**

Introduction to designs – Industrial Designs, Design Laws in India: Designs Act of 2000, Registration of Design, Owners Rights, Piracy of Designs, Offence, Remedies and Enforcement, Designs- International Law, Introduction to Geographical Indication, Law of Geographical Indication in India: Geographical Indication of Goods (Registration and Protection) Act, 1999, Register of Geographical Indication, Infringement of Registered Geographical Indication Offence, Remedies and Enforcement

UNIT 4 :**Introduction to Cyber Law & Information Technology Act, 2000**

Evolution of Computer Technology, Emergence of Cyberspace, Cyber Jurisprudence, Jurisprudence and Law, Doctrinal Approach, Consensual Approach, Real Approach, Cyber Ethics, Cyber- Jurisdiction, Hierarchy of Courts, Civil and Criminal Jurisdictions, Cyberspace –Web space (WWW), Web Hosting and Web Development Agreements (specimen), Domain Names, Internet as a Tool for Global Access, Overview of IT Act, 2000, Amendments and Limitations of IT Act,

UNIT 5

Digital Signatures, Cryptography, Cryptographic Algorithm, Public Cryptography, Private Cryptography, Electronic Governance, Legal Recognition of Electronic Records, Legal Recognition of Digital Signature, Certifying authorities, Cyber Crime and Offences, Network Service Provider Liability, Cyber Regulation Appellate tribunal, Penalties and Adjudication

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**UNIT 6 :****Cyber law issues and related legislation**

Patent Laws, Trademark law, Copyright, Software –copyright or patented, Domain Name and Copyright disputes, Electronic Database and its Protection, IT Act and Civil procedure Code, IT Act and Criminal procedure Code, Relevant Sections of Indian Evidence Act, Relevant Sections of Bankers Book Evidence Act, Relevant Section of Indian penal Code, Relevant Section of Reserve Bank of India Act, Law Relating to Employees and Internet, Alternative Dispute resolution, Online Dispute Resolution (ODR)

TEXT BOOKS:

1. Law and practice of intellectual property in India by Vikas Vashishth
2. Intellectual property by A.Kalank
3. Intellectual property- patents,copyrights,trade marks and allied rights by Cornish W R
4. Patents ,copyrights, trade marks and design by B L Wadhera
5. Intellectual property law by P Narayana
6. Patents ,copyrights, trade marks and design by Rajeev Jain

(A0539128) GREEN IT
(OPEN ELECTIVE IV)**COURSE OBJECTIVES**

At the end of this course students will be able to:

- Explain the benefits of adopting Green IT for both the IT service provider and the organization.
- Conduct an Internal assessment of their organization: where are we now?
- Describe how to assess an Organization's business operations, IT use and products/services in terms of their energy consumption and understand the associated Primary and Secondary carbon footprint.

OUTCOMES:

Candidates can expect to have gained knowledge and understanding of the following areas:

1. An overview of Green IT
2. Internal assessment of your organisation
3. Improving your organisation's Green IT credentials

UNIT - I

What is Green IT? An Overview: Understand the overall need for an organisation to adopt a Green IT strategy: The reality of climate change and over-population - A brief understanding of climate change science - Consumption of precious resources such as oil, gas and water - The effect of poor and unsustainable working practices - The effect of hazardous waste use and disposal. **Provide an understanding of the historic development and context of the Kyoto –Protocol:-** The Kyoto Protocol – and historic background - The formation of the International Panel on Climate Change - The Rio Earth Summit - The UN Framework Convention on Climate Change and IPCC - Further rounds of COP (Copenhagen, Cancun, etc.)

UNIT -II

Understand the definition of 'Green IT'.- The key elements of Green IT - Commonly accepted -definitions - IT as an energy consumer - IT as a green 'enabler' - The concept and dangers of 'Green Wash' . **Identify and understand an organisation's external drivers and opportunities for greening its IT-** Political drivers - Environmental drivers - Social drivers - Legal drivers - Economic drivers .
Identify and understand the internal drivers, opportunities and benefits of adopting aGreen IT strategy for both an organisation and its IT service provider(s) - Cost - Operations - Marketing/PR - Culture

UNIT - III

Internal assessment of your organisation: where are we now?

Gain an understanding of how to create an Green IT policy : The role of a Green IT policy - The importance of a Green IT policy - The alignment of Green IT policy with an organisation's environmental, - sustainability and Corporate Social Responsibility policies - How to overcome the dangers of 'Green Wash'

Know how to assess an organisation's business operations; in terms of their carbon

footprint.- Definition of carbon footprints: direct and indirect emissions - Examples of an organisation's footprints: direct and indirect emissions - An understanding of carbon emissions across a product/service lifecycle including: 1. Concept & design 2. Material extraction 3.Transport 4. Manufacture 5. Usage 6.Disposal
Carbon Footprint Calculators Carbon Offsetting and Carbon Neutrality Carbon trading

UNIT III

Understand the contribution that emissions from the use of IT is making to those

carbon footprints in terms of energy consumption and behaviours. PCs and mobile devices - Office applications and equipment - Communication and collaboration technologies - Servers - Data storage - Data centres. **Describe how to audit an organisation's existing IT functions and processes.** Identification of energy and carbon inefficiencies- Planning and prioritising green IT initiatives - Establishing a continuous improvement framework for Green ICT including use of the ITIL Continual Service Improvement Model - Tools and methods available - Roles and responsibilities

UNIT -IV

Understand the importance and risks, issues and opportunities around improving

efficiency. - Printing and recycling - Desktop kit (monitors, processors, external devices) - Mobile and remote communications - Server Rooms and Data Centres Including approaches for: Assessing environmental and

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

property impacts eg heating/cooling - Device consolidation - Different approaches to CPU, server and data storage optimisation - Thin clients - Virtualisation eg servers, disk tiering - Dematerialisation – from assets to services - Cloud computing/Software as a Service (SaaS) - Power provisioning and management

UNIT V**Understand the concept of total systems lifecycle management that supports IT assets**

from manufacture to disposal and its carbon impact: The supply chain for IT products and services -The impact of suppliers and customers - Procuring for Green IT, including: -1. Supplier selection criteria 2. Working with suppliers 3. Tools and methods available such as Suppliers and Contracts database 4. the benefits . Providing efficient, low carbon support and maintenance, including: 1. Service desk 2. Change management 3. Service Asset and Configuration Management 4. Training

Understand how best to re-use, recycle and dispose of IT assets: The impact of equipment refresh cycles - Different approaches to product disposal - Tools and approaches available - The benefits of re-use, recycle and dispose. **Developing a Green IT Action Plan:** The scope - Time lining and budgeting - Roles and responsibilities - Tools and methods - Monitoring and measurement - The benefits - Employee engagement and management - Stakeholder identification, engagement and management

UNIT VI**Deployment of ICT for sustainability across your organisation’s activities**

Understand how to embed the use of ICT for reducing emissions from business activities, in an IT Strategy for Sustainability: The role of IT in: their business units product/service delivery - achieving wider business goals - investments to achieve carbon reductions - achieving carbon neutrality **Discuss the roles and responsibilities associated with Green IT and IT for sustainable**

Operations: The Green IT Champion - Procurement Manager - Sustainability Officer - Corporate Social Responsibility (CSR) Manager .**Explain how to encompass Green IT and IT for sustainable operations in end-to-end lifecycle costing, business cases and TCO:** New accounting practices, to include ‘social’ accounting methods such as Triple Bottom Line - Definition and calculation of end-to-end lifecycle costs - Definition and calculation of payback periods - Definition and calculation of Total Cost of Ownership - Tools and methods available

TEXT BOOK:

1. Green it for sustainable business practice, an ISEB Foundation Guide Mark O’Neill, BCS Publishing

REFERENCES:

1. Green it in practice Gary Hird, IT Governance Publishing
2. Green it, reduce your information system’s environmental impact While adding to the bottom line Toby J Velte, Anthony T Velte, Robert Elsenpeter, McGraw Hill Publishing
3. the Green and Virtual data Center Greg P Schulz, CRC Press

(A0540128) COMPUTER FORENSICS

(OPEN ELECTIVE IV)

OBJECTIVES:

- This course is intended to provide students with greater depth of study in a number of key topics in the area of computer security in society: Cyber crime, computer and forensics, analysis. The course also involves significant practical work involving the development and use of forensics analysis tools.

OUT COMES:

- Understand financial and accounting forensics, and explain their role in preventing various forms of fraud.
- Distinguish various types of computer crime, and use computer forensic techniques to identify the digital fingerprints associated with criminal activities.
- Know how to apply forensic analysis tools to recover important evidence for identifying computer crime.
- Be able to develop a custom computer forensic analysis tool.

UNIT-1

Computer Forensics Fundamentals: Introduction to Computer Forensics, use of Computer Forensics in Law Enforcement, Computer Forensics Assistance to Human Resources/ Employment Proceedings, Computer Forensics Services, Benefits of Professional Forensics Methodology, Steps Taken by Computer Forensics Specialists.

Types of Computer Forensics Technology: Types of Military Computer Forensic Technology, Types of Law Enforcement: Computer Forensic Technology, Types of Business Computer Forensic Technology.

UNIT-2

Computer Forensics Evidence and Capture: Data Recovery Defined, Data Backup and Recovery, The Role of Backup in Data Recovery, The Data-Recovery Solution.

Evidence Collection and Data Seizure: Why Collect Evidence?, Collection Options, Obstacles, Types of Evidence, The Rules of Evidence, Volatile Evidence, General Procedure, Collection and Archiving, Methods of Collection, Arty facts, Collection Steps, Controlling Contamination: The Chain of Custody.

UNIT-3

Duplication and Preservation of Digital Evidence: Preserving the Digital Crime Scene, Computer Evidence Processing Steps, Legal Aspects of Collecting and Preserving Computer Forensic Evidence

Computer Image Verification and Authentication: Special Needs of Evidential Authentication, Practical Considerations Practical Implementation.

UNIT-4

Processing crime and incident scenes: Identifying digital evidence, collecting evidence in private-sector incident scenes, Processing law enforcement crime scenes, preparing for a search, securing a computer incident or crime scene, seizing digital evidence at the scene, storing digital evidence, obtaining a digital hash.

UNIT-5

Current computer forensics tools: Evaluating computer forensics tool needs, computer forensics software tools, computer forensics hardware tools, validating and testing forensics software.

UNIT-6

E-mail investigations: Exploring the role of e-mail in investigation, exploring the roles of the client and server in e-mail, investigating email crimes violations.

Cell phone and Mobile device forensics: understanding mobile device forensics, understanding acquisition procedures for cell phones and mobile devices.

TEXT BOOKS:

1. "Computer forensics, computer crime scene investigation" by John Vacca CHARLES RIVER MEDIA -second edition.
2. Guide to "Computer forensics and investigations" by Bill Nelson, Amelia Phillips and Christopher Stuart, CENGAGE- third edition.

REFERENCES:

1. Real digital forensics by Keith J Jones, Richard Bejtlich, Curtis Wrose-pearson education.
2. Software forensics collecting evidence from a scene of a digital crime by Robert M Slade TMH2005

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. II-Sem (CSE)

T	C
3+1*	3

(A0541128)CONCEPTS OF MACHINE LEARNING

(Elective V)

OBJECTIVES:

To give students

- Basic knowledge about the key algorithms and theory that form the foundation of machine learning and computational intelligence
- A practical knowledge of machine learning algorithms and methods

OUT COMES:

They will be able to

- Understand the principles, advantages, limitations and possible applications of machine learning
- Identify and apply the appropriate machine learning technique to classification, pattern recognition, optimization and decision problems.

UNIT – I**Introduction:**Machine perception, pattern recognition example, pattern recognition systems, the design cycle, learning and adaptation.**UNIT – II****Bayesian Decision Theory:**Introduction, continuous features – two categories classifications, minimum error-rate classification- zero–one loss function, classifiers, discriminant functions, and decision surfaces.**UNIT – III****Maximum likelihood and Bayesian parameter estimation:** Introduction, maximum likelihood estimation, Bayesian estimation, Bayesian parameter estimation–Gaussian case.**UNIT – IV****Un-supervised learning and clustering:** Introduction, mixture densities and identifiability, maximum likelihood estimates, application to normal mixtures, K-means clustering. Data description and clustering – similarity measures, criteria function for clustering.**UNIT – V****Component analysis:** Principal component analysis, non-linear component analysis; Low dimensional representations and multi-dimensional scaling.**UNIT-VI****Stochastic methods:** Introduction, Stochastic search, Boltzmann learning**TEXT BOOKS:**

1. “**Pattern classifications**”, Richard O. Duda, Peter E. Hart, David G. Stroke. Wiley student edition, Second Edition.

REFERENCE BOOKS:

1. “**Pattern Recognition and Image Analysis**” – Earl Gose, Richard John baugh, Steve Jost.
2. “**Introduction to Machine Learning**” by Ethem Alpaydin, PHI 2nd Edition.

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. II-Sem (CSE)

T	C
3+1*	3

(A0542128) GAME THEORY

(ELECTIVE V)

OBJECTIVES

- To give an overview of a broad range of models that are studied in game theory.
- To discuss the main concepts in the game theory.
- To explain the classes of games.
- To study the mathematics associated to zero-sum games.
- To discuss the application of game theory.

OUTCOMES

- Knowledge and understanding .
 - a) Be familiar with the main concepts on game theory.
 - b) Use the Minimax (Maximin) criterion.
 - c) Derive the classes of games.
 - d) Use matrix games .
- Cognitive skills (thinking and analysis).
 - a) To identify and solve problems in real-life . Work with given information ,form games classes and try to solve them.
- Communication skills (personal and academic).
 - a) Encourage the students to be self starters (creativity, decisiveness, initiative)and to finish the mathematical problems properly (flexibility, adaptability).
 - b) Also to improve general performance of students through the interaction with each other in solving different game problems.

UNIT -I

Introduction and Outline of the Course, Definitions, Utilities, Rationality, Intelligence, Common knowledge, Classification of Games

NON-COOPERATIVE GAME THEORY: Extensive Form Game, Strategic Form Games with Illustrative Examples, Dominant Strategy Equilibria

UNIT –II

Pure Strategy Nash Equilibrium with Illustrative Examples and Key Results, Mixed Strategy Nash Equilibrium with Illustrative Examples and Key Results such as the Nash Theorem, Computation of Nash Equilibria and introduction to algorithmic theory, Matrix Games: Saddle Points, Minimax Theorem

UNIT- III

Bayesian Games, Bayesian Nash Equilibrium, Evolutionary Game Theory (ESS Strategies)

Repeated Game , **MECHANISM DESIGN** The Mechanism Design Environment , Social Choice Functions with Illustrative Examples, Implementation of Social Choice Functions

UNIT IV

Incentive Compatibility and Revelation, Theorem, Gibbard-Satterthwaite and Arrow, Impossibility Theorem, Vickrey-Clarke-Groves (VCG) Mechanisms, Bayesian Mechanisms (dAGVA) , Revenue Equivalence Theorem

UNIT V

Myerson Optimal Auction , Further Topics in Mechanism Design, **COOPERATIVE GAME THEORY** Correlated Strategies and Correlated, Equilibrium, The Nash Bargaining Problem

UNIT VI

Coalitional Games (Transferable Utility Games), The Core, The Shapley Value , Other Solution Concepts: Kernel, Nucleolus , To Probe Further and Conclusion of the Course

TEXT BOOKS:

1. Y. Narahari, Dinesh Garg, Ramasuri Narayanam, Hastagiri Prakash Game Theoretic Problems in Network Economics and Mechanism Design Solutions. Springer Series in Advanced Information and Knowledge Processing (AIKP), London, UK, 2009. (URL:<http://www.springer.com/math/applications/book/978-1-84800-937-0>)
2. Roger B. Myerson. Game Theory: Analysis of Conflict. Harvard University Press, September 1997.
3. NPTEL Game Theory Course

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. II-Sem (CSE)

T	C
3+1*	3

(A1225126) MULTIMEDIA AND APPLICATION DEVELOPMENT

(ELECTIVE V)

(Common to CSE&IT)

COURSE OBJECTIVES:

1. The skills have to be acquired across a range of careers in multimedia including games development, web application development, multimedia application development, computer based training, e-Learning application development, interactive multimedia development, software development, software analysis, project management.

LEARNING OUTCOMES:

The student will be able to:

1. Have gained a thorough understanding of the technical design and implementation issues involved in building complex multimedia systems or web-based systems/applications.
2. Apply appropriate design solutions and usability principles in their multimedia systems applications.
3. Analyse how digital media and technology impacts on society.

UNIT I: FUNDAMENTAL CONCEPTS IN TEXT AND IMAGE

Multimedia and hypermedia, world wide web, overview of multimedia software tools. Graphics and image data representation graphics/image data types, file formats, Color Models in images.

UNIT II: FUNDAMENTAL CONCEPTS IN VIDEO AND DIGITAL AUDIO

Types of video signals, analog video, digitization of sound, quantization and transmission of audio.

UNIT III: ACTION SCRIPT I

ActionScript Features, Object-Oriented ActionScript, Datatypes and Type Checking, Action Script Classes:Defining classes, Constructor functions, Properties, Methods, Constructors, Completing Box Class.

UNIT : ACTION SCRIPT II

Inheritance, Interfaces, Packages, Exceptions.

UNIT V: APPLICATION DEVELOPMENT

An OOP Application Frame work, Using Components with ActionScript 2.0.

UNIT VI: FUNDAMENTALS OF MULTIMEDIA DATA COMPRESSION

Lossless compression algorithm:introduction,Basics of Information Theory, Run-Length Coding, Variable Length Coding, Dictionary Based Coding, Arithmetic Coding, Lossless Image Compression

Lossy compression algorithm: Introduction, Distortion Measures, The rate distortion Theory, Quantization.

TEXT BOOKS:

1. Fundamentals of Multimedia by Ze-Nian Li and Mark S. First Edition, Drew PHI/Pearson Education,2003.
2. Essentials ActionScript 2.0, Colin Mook, 2nd Edition, SPD O,REILLY, 2004.

REFERENCE BOOKS:

1. Digital Multimedia, Nigel chapman and jenny chapman, Wiley-Dreamtech.Macromedia Flash MX Professional 2004 Unleashed, Pearson.
2. Multimedia and communications Technology, Steve Heath, Elsevier (Focal Press).
3. Multimedia Applications, Steinmetz, Nahrstedt, Springer.
4. Multimedia Basics by Weixel Thomson
5. Multimedia Technology and Applications, David Hilman ,Galgotia

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. II-Sem (CSE)

T	C
3+1*	3

(A0543128) ADHOC AND WIRELESS SENSOR NETWORKS

(ELECTIVE V)

COURSE OBJECTIVES

- Knowledge of mobile ad hoc networks, design and implementation issues, and available solutions.
- Knowledge of routing mechanisms and the three classes of approaches: proactive, on-demand, and hybrid.
- Knowledge of clustering mechanisms and the different schemes that have been employed, e.g., hierarchical, flat, and leaderless.
- Knowledge of the 802.11 Wireless Lan (WiFi) and Bluetooth standards. This includes their designs, operations, plus approaches to interoperability.
- Knowledge of sensor networks and their characteristics. This includes design of MAC layer protocols, understanding of power management, query processing, and sensor databases.
- Hands-on experience in designing and implementing ad hoc network functionality using network simulation tools and Pocket PCs.

OUTCOMES

- Have an understanding of the principles of mobile ad hoc networks (MANETs) and what distinguishes them from infrastructure-based networks.
- Have an understanding of the principles and characteristics of wireless sensor networks (WSNs).
- Understand how proactive routing protocols function and their implications on data transmission delay and bandwidth consumption.
- Understand how reactive routing protocols function and their implications on data transmission delay and bandwidth consumption.
- Understand how hybrid routing protocols function and their ability to balance speed and bandwidth consumption.
- Understand how the different clustering algorithms and their usefulness for network management and routing.

UNIT-1

Introduction to ad hoc wireless networks, characteristics of MANETs, applications of MANETs

UNIT-2**Routing in MANETs:** topology based and position based approaches, topology based routing protocols, position based routing protocols, other routing protocols**UNIT-3****Data transmission in MANETs:** the broadcast storm, multicasting, geo casting**UNIT-4****Wireless LANs:** Introduction, why wireless LANs, transmission techniques, medium access control protocol issues, the IEEE 802.11 standards for wireless LANs**UNIT-5****Wireless PANs:** Introduction, why wireless PANs?, the Bluetooth technology, enhancements to Bluetooth, IEEE 802.15 working group for WPANs**Wireless mesh networks:** Introduction: network architecture, challenging techniques**UNIT-6****Security:** Security in ad hoc wireless networks, key management, secure routing cooperation in MANETs, intrusion detection systems**TEXT BOOKS:**

1. Ad hoc and sensor networks- theory and applications 2nd edition, Dharma prakash agrawal, World scientific publishing co.pte.ltd

REFERENCES:

1. Ad hoc wireless networks- architectures and protocols, C.Sivaram murthy, B.S.Murthy, Pearson education, 2004
2. Security in ad hoc sensor networks, Rahim, Beyah, et al, old scientific publications

AUTONOMOUS

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

IV B.Tech. II-Sem (CSE)

T	C
3	2

(A0015125) MAN MANAGEMENT**(Skill Development Course-3*)**

(Common to CSE &IT)

COURSE OBJECTIVE:

1. To inculcate Human Values and ethos and to introduce to the cultural heritage of Bharath.
2. To produce IT personal committed to the creation of a happier society through IT.

LEARNING OBJECTIVES:

After going through this course students will be able to

1. Appreciate the importance of values in day to day life.
2. Appreciate the great Indian heritage and what it has taught over generations
3. Apply some these both at work and at home.

UNIT I:

LIFE STYLE OF A LEADER-MANAGER-I: The Manager and Values-Based Management: Significance and Salient Features, Individual Character and National Character: Fundamental Requisites for an Ideal Leader-Manager [Chapters 9 & 10 from the book "Man Management"]

UNIT II:

LIFE STYLE OF A LEADER-MANAGER-II: Guidelines for a Values-Based Leader-Manager: Spiritual, Personal and Business, Ideal Qualities of a Leader-Manager: Insights from Indian Scriptures [Chapters 11 & 12 from the book "Man Management"]

UNIT III:

FACTORS OF EXCELLENCE IN MANAGEMENT: INDIVIDUAL FACET-I: Personality and Its Influencing Factors, Manager and Self-Concept: An Indian Perspective, In Search of Excellence in Man-Management: An Indian Perspective [Chapters 13, 14 & 15 from the book "Man Management"]

UNIT IV:

FACTORS OF EXCELLENCE IN MANAGEMENT: INDIVIDUAL FACET-II: Spiritual Insights for Man-Management, Manager and Mind Control, WATCH Method of Transformational Leadership. [Chapters 16, 22 & 23 from the book "Man Management"]

UNIT V:

VALUES-BASED BUSINESS AND MANAGEMENT: CONCEPTUAL BACKGROUND-I: Art and Science of Management in Ancient India, the Life Breaths of Business Management and Organisation. [Chapters 4 & 5 from the book "Man Management"]

UNIT VI:

VALUES-BASED BUSINESS AND MANAGEMENT: CONCEPTUAL BACKGROUND-II: Role of Values In Management Education, Roles and Responsibilities of Business Executives in the contemporary World, Need and Significance of Morality and Ethics in the World of Business And Finance.(Chapters 24,25 & 26 from the book "Man Management"]

TEXT BOOKS:

1. "Man Management- A Value-Based Management Perspective", Based on The Discourses of Sri Sathya Sai Baba, Compiled by Prof. Racherla Kumar Bhaskar ,Published by The Sai Publications Division, Sri Sathya Sai Students And Staff Welfare Society, Sri Sathya Sai Vidyagiri, Prasanthinilayam, August 2012[ISBN:978-81-909128-2-2]

REFERENCE BOOKS:

2. References from the book "Man Management", compiled by Prof. Racherla Kumar Bhaskar