FACULTY OF ENGINEERING

IASE Deemed University

Gandhi VidyaMandir

Sardarshahr (Rajasthan) – 331401

INDIA

Teaching and Examination Scheme and Syllabus

for

BACHELOR OF TECHNOLOGY

(Four-Year Full Time Degree Programme)

COMPUTER SCIENCE ENGINEERING

2ND TO 4TH YEAR

(SEMESTER SCHEME)

Sessions 2016-17, 2017-18, 2018-19

RULES AND GUIDELINES FOR THE STUDENTS

1. **The Bachelor of Technology (Computer Science & Engineering) course is a four year (Eight Semester) full time integrated degree programme**.

2. **THE PROGRAMME**

The Bachelor of Technology (Computer Science & Engineering) is a four year (Eight semesters) full time degree programme .The course structure and programme administration are as follows.

3. **COURSE STRUCTURE**

The four year, eight semester teaching comprises of Theory (Lectures and Tutorials) and Practicals/Sessionals (Laboratory work, Engineering Graphics, Workshop Practice and Project etc.).Examination will be held at the end of the each semester. Details of these are given in the Teaching & Examination Scheme.

4. **PROGRAMME ADMINISTRATION**

4.1. Medium of Instruction

English shall be the medium of instruction and examination.

4.2. EVALUATION

(a) Each subject will be evaluated through a theory paper at the end of the semester carrying 80 marks along with continuous evaluation of sessional work, carrying 20 marks. The theory paper shall be of three hour duration. The sessional work will consist of continuous assessment of student's performance by teachers in tutorial classes, and class tests.

(b) Three class tests will be organized in each semester as per the scheme. The higher two out of the marks scored in the three tests will be considered for the sessional marks.

(c) Evaluation of laboratory practical work and Engineering Graphics (Drawing) will be through continuous assessment throughout the semester as well as examination at the end of the semester.

(d) At the end of the sixth semester the student will undergo practical training for a period of at least 45 working days in an industry / research organization related to his / her field of Study. At the end of the training, the student will submit its report to the Head of the Department within three weeks of the start of the seventh semester. The work of the practical training will be evaluated by a board of two teachers appointed by the Head of the Department. The later will counter sign the marks awarded by the board.

(e) Project: The project work will be carried out in the VII & VIII semester. The topic of the project will be approved by the Head of the Department and the entire project work will be carried out under the guidance of a teacher of the department approved as project supervisor by the Head of the Department. The nature of the project work will consist of varying proportions of designing, fabrication, testing and

analysis of results. The project topic can also be taken from a live industrial problem. The report of the completed project shall be signed by the guide and submitted to the Head of the Department on or before the last working day of the eighth semester. The evaluation of the project will be done by a board consisting of two examiners.

5. Promotion

5.1 The maximum span period of a programme is eight years from the date of registration in the programme.

5.2 The minimum grade for passing the examination for each semester shall be **"P"** of all the subjects (theory, sessional) of the semester.

5.3 A student will be permitted to attend the classes of the fourth/sixth/eighth semesters immediately after the examination of the third/fifth/seventh semester's examination, as the case may be, provided he/she has appeared in the first/third/fifth/seventh semester examination, respectively.

5.4 To be eligible for promotion to the 5^{th} semester of the programme a student must have successfully cleared 50% of the total subjects including practicals and sessionals of the third and fourth semesters taken together. In case of 50% of total number of papers is fractional number, the candidate must have cleared number of papers next higher number of the fraction so obtained.

5.5 To be eligible for promotion to the 7^{th} semester of the programme a student must have successfully cleared 50% of the total subjects including practicals and sessionals of the fifth and sixth semesters taken together. In case of 50% of total number of papers is fractional number, the candidate must have cleared number of papers next higher number of the fraction so obtained.

5.6 A student promoted to the third/fifth/seventh semesters, without having cleared all the papers, will have to appear and pass the backlog papers of the first/third/seventh semesters along with the regular examination of the first/third/fifth semesters and backlog papers of the second/fourth/sixth semesters along with the regular examination of the second/fourth/sixth semesters.

5.7 (a) Award of Grade:

- **Academic Year**: Two consecutive (one odd + one even) semesters constitute one academic year.
- **Choice Based Credit System (CBCS):** The CBCS provides choice for students to select from the prescribed courses (core, elective courses).
- **Course**: Usually referred to, as 'papers' is a component of a programme. All courses need not carry the same weight. The courses should define learning objectives and learning outcomes. A course has been designed to comprise lectures/ tutorials/laboratory work/ field work/ outreach activities/ project work/ vocational training/viva/seminars/term papers/assignments/presentations/self-study etc. or a combination of some of these.

- **Credit Based Semester System (CBSS)**: Under the CBSS, the requirement for awarding a degree is prescribed in terms of number of credits to be completed by the students.
- **Credit Point**: It is the product of grade point and number of credits for a course.
- **Credit**: A unit by which the course work is measured. It determines the number of hours of instructions required per week. One credit is equivalent to one hour of teaching (lecture or tutorial) or two hours of practical work/field work per week.
- **Cumulative Grade Point Average (CGPA)**: It is a measure of overall cumulative performance of a student over all the semesters. The CGPA is the ratio of total credit points secured by a student in various courses in all semesters and the sum of the total credits of all courses in all the semesters. It is expressed up to two decimal places.
- **Grade Point**: It is a numerical weight allotted to each letter grade on a 10-point scale.
- **Letter Grade:** It is an index of the performance of students in a said course/semester/programme. Grades are denoted by letters O, A+, A, B+, B, C, P and F.

Letter Grade	% Scale	Grade Point
O (Outstanding)	85% and Above	10
A+(Excellent)	75% to 84.99%	9
A(Very Good)	65% to 74.99%	8
B+(Good)	55% to 64.99%	7
B(Above Average)	50% to 54.99%	6
C(Average)	45% to 49.99%	5
P (Pass)	40% to 44.99%	4
F(Fail)	Less than 40%	0
Ab (Absent)	0 %	0

- **Programme**: An educational programme leading to award of a Degree.
- **Semester Grade Point Average (SGPA)**: It is a measure of performance of work done in a semester. It is ratio of total credit points secured by a student in various courses registered in a semester and the total course credits taken during that semester. It shall be expressed up to two decimal places.
- **Semester**: Each semester will consist of 13-18 weeks of academic work equivalent to 90 teaching days. The odd semester may be scheduled from July to December and even semester from January to June.

- **Transcript or Grade Card or Certificate:** Based on the grades earned, a grade certificate shall be issued to all the registered students after every semester. The grade certificate will display the course details (code, title, credit points, grade secured) along with SGPA of that semester and CGPA earned till that semester.

5.8 If a student (who has successfully completed the programme) wishes to reappear in one or more theory papers of the first, second, third, fourth, fifth, sixth, seventh or eighth semesters for the purpose of improving his/her marks, he/she will be permitted to do so on payment of requisite examination fee along with the regular examinations of that semester; however, the total number of such attempts shall not exceed four theory papers during the span period of the programme. For this his/her previous performance in the paper/papers concerned shall be treated as cancelled. The application for such reappearing/re-examination must be submitted before the next examination of the corresponding semester. However, such candidates shall not be considered for award of gold medal.

5.9 A student to be eligible for award of degree has to clear all papers offered during four-year programme within the span period of eight years.

6. LATERAL ENTRY

Students who have passed 3 year diploma examination from the Board of Technical Education, Rajasthan, or its equivalent with a minimum of 60% marks can be admitted to the Third Semester of the B. E. programme. However, they will be required to pass a course on Special Mathematics (BE300) for Diploma pass students. Students will have to pass this course before they are admitted to the seventh semester. However, the marks obtained in this course will not be counted for deciding the division of the student.

7. Attendance: All students are required to have 75% attendance in each subject and there must be 75% attendance of the student before he/she could be permitted to appear in the examination.

8. RULES FOR CHANGE OF BRANCH FOR THE STUDENTS OF III SEM. B.TECH/ B.E.:

I The faculty, on the basis of applications received from desirous students up to the date and time notified by the Director, will prepare a merit list of the students. The list will be prepared on the basis of overall merit of the I(Semester) result only and the applications for change of branch will be processed as per the merit list.

ELIGIBILITY CRITERIA:

(a) The students must have passed the I Semester B.Tech. Examination in all components in one attempt with at least **"B+"** grade. The student with back papers or whose result has not been declared will not be considered for change of branch.

(b) In case any student has applied for re-valuation/ re-totalling of his/her marks of I Semester B.Tech and the result has not been received till the time of change of branch, such a student

will not be entitled for change of branch on the basis of his/her subsequently revised result.

PROCEDURE:

1) Applications in a specified format (developed by the faculty) for change of branch will be invited by the Director/Principal of the faculty on the basis of the result of I (Semester) B. Tech. in duplicate, upto the date notified by IASE University. One copy of each such application be sent to IASE University by that date.

2) The students would submit a photo copy of I (Semester) Examination mark sheet of that year along with the application. The student may give as many preferences as possible against the vacant seats in respective college.

3) A seat matrix shall be prepared by the faculty, as per the details of the vacant seats (admitted through direct admission) in the previous year.

4) Due to change of branch, the strength of student in any branch should not fall short of 75% of the enrolled students in that branch in that year. And under no circumstances, due to change of branch, the number of seats in a particular branch in a college shall exceed the sanctioned strength approved by the AICTE, for that batch.

5) All students who have applied for the change of branch in-time will be called for counselling by the admission council of the faculty and considered for change of branch as per merit, preference and availability of seat. However, at the time of the counselling, if any student wishes to withdraw his/her application he/she can do so by a written request. In case any student does not present himself/herself for counselling, his/her branch will be changed as per the preference mentioned in the application form, merit and availability of seat. Once a student has been permitted to change of a branch it will not be withdrawn.

TEACHING & EXAMINATION SCHEME

FOR B.TECH- FOUR YEAR (8 SEMESTER) FULL TIME DEGREE

B.TECH COMPUTER SCIENCE&ENGINEERING SECOND YEAR

SEMESTER: III

Subject Code	Title	Hrs. / Week			Credit	IA		Exam		Total
		L	T	P		Th	P	Th	P	
CSE301&301-P	Electronic Devices and Circuits	3	1	2	5	20	45	80	30	175
CSE302&302-P	Data Structures and Algorithms	3	1	2	5	20	45	80	30	175
CSE303&303-P	Digital Electronics	3	1	2	5	20	45	80	30	175
CSE304&304-P	Linux and Shell Programming	3	-	2	4	20	45	80	30	175
CSE305&305-P	Object Oriented Programming	3	-	2	4	20	30	80	20	150
CSE306	Advanced Engineering Mathematics	3	1	-	4	20	-	80	-	100
CSE 307	Discipline & Extra Curricular Activity									50
	Total	**18**	**4**	**10**	**27**					**1000**

SEMESTER: IV

Subject Code	Title	Hrs. / Week			Credit	IA		Exam		Total
		L	T	P		Th	P	Th	P	
CSE401&401-P	Microprocessors and Interfaces	3	1	2	5	20	45	80	30	175
CSE402	Dicrete Mathematical Structures	3	1	-	4	20	-	80	-	100
CSE403	Statistics and Probability Theory	3	1	-	4	20	-	80	-	100
CSE404&404-P	Digital Logic Design	3	1	2	5	20	60	80	40	200
CSE405&405-P	Principles of Communication	3	1	2	5	20	60	80	40	200
CSE406	Principles of Programming Languages	3	-	-	3	20	-	80	-	100
CSE407-P	Business Entrepreneurship Development	-	-	2	1	-	45	-	30	75
CSE 408	Discipline & Extra Curricular Activity									50
	Total	**18**	**5**	**08**	**27**					**1000**

IA- Internal Assessment **T- Tutorial**

L- Lecture **P- Practical**

Th- Theory

TEACHING & EXAMINATION SCHEME

FOR B.TECH- FOUR YEAR (8 SEMESTER) FULL TIME DEGREE

B.TECH COMPUTER SCIENCE&ENGINEERING THIRD YEAR

SEMESTER: V

Subject Code	Title	Hrs. &Week			Credit	IA		Exam		Total
		L	T	P		Th	P	Th	P	
CSE501	Computer Architecture	3	1	-	4	20	-	80	-	100
CSE502&502-P	Software Engineering	3	1	2	5	20	45	80	30	175
CSE503	Theory Of Computation	3	1	-	4	20	-	80	-	100
CSE504&504-P	Database Management Systems	3	1	2	5	20	60	80	40	200
CSE505&505-P	Operating Systems	3	1	2	5	20	60	80	40	200
CSE506 I	Advanced Data Structure	3	-	-	3	20	-	80	-	100
CSE506 II	Digital Signal Processing									
CSE506 III	Information Theory & Coding									
CSE507-P	System Design in UML Lab.	-	-	2	1	-	45	-	30	75
CSE 508	Discipline & Extra Curricular Activity									50
	Total	**18**	**5**	**8**	**27**					**1000**

SEMESTER: VI

Subject Code	Title	Hrs. & Week			Credit	IA		Exam		Total
		L	T	P		Th	P	Th	P	
CSE601	Computer Networks	3	-	-	3	20	-	80	-	100
CSE602&602-P	Design and Analysis of Algorithms	3	1	2	5	20	45	80	30	175
CSE603	Telecommunication fundamental	3	1	-	4	20	-	80	-	100
CSE604&604-P	Computer Graphics and MultimediaTechniques	3	1	2	5	20	45	80	30	175
CSE605&605-P	Embedded System Design	3	1	2	5	20	45	80	30	175
CSE606 I	Advance Topics in Operating Systems	3	-	-	3	20	-	80	-	100
CSE606 II	Artificial Intelligence									
CSE606 III	Human Computer Interface									
CSE607-P	Java Programming Lab	-	-	2	1	-	45	-	30	75
CSE608-P	Humanities and Social Sciences	-	-	2	1	-	30	-	20	50
CSE 608	Discipline & Extra Curricular Activity									50
	Total	**18**	**4**	**10**	**27**					**1000**

IA- Internal Assessment **T- Tutorial**

L- Lecture **P- Practical**

Th- Theory

TEACHING & EXAMINATION SCHEME

FOR B.TECH- FOUR YEAR (8 SEMESTER) FULL TIME DEGREE

B.TECH COMPUTER SCIENCEENGINEERING FOURTH YEAR

SEMESTER: VII

Subject Code	Title	Hrs. & Week			Credit	*IA*		*Exam*		Total
		L	T	P		Th	P	Th	P	
CSE701	Cloud Computing	3	1	-	4	20	-	80	-	100
CSE702	Information System Security	3	-	-	3	20	-	80	-	100
CSE703	Data Mining & Ware Housing	3	1	-	4	20	-	80	-	100
CSE704&704-P	Computer Aided Design for VLSI	3	1	2	5	20	45	80	30	175
CSE705&705-P	Compile Design	3	1	2	5	20	45	80	30	175
CSE706 I	Advance Data Base ManagementSystems	3	-	-	3	20	-	80	-	100
CSE706 II	Robotics									
CSE706 III	Data Compression Techniques									
CSE707-P	Web Development Lab	-	-	2	1	-	30	-	20	50
CSE 708-P	Project-1	-	-	2	1	-	50	-	-	50
CSE 709-P	Practical Training*	-	-	2	1	-	-	-	100	100
CSE 710	Discipline & Extra Curricular Activity									50
	Total	**18**	**4**	**10**	**27**					**1000**

SEMESTER: VIII

Subject Code	Title	Hrs. & Week			Credit	*IA*		*Exam*		Total
		L	T	P		Th	P	Th	P	
CSE801	Mobile Computing	3	1	-	4	20	-	80	-	100
CSE802&802-P	Digital Image Processing	3	1	2	5	20	60	80	40	200
CSE803	Distributed Systems	3	1	-	4	20	-	80	-	100
CSE804 I	Hardware Testing & Fault Tolerance	3	1	-	4	20	-	80	-	100
CSE804 II	Real Time Systems									
CSE804 III	Information Retrieval									
CSE 805-P	Unix Network Programming & Simmulation Lab	-	-	2	1	-	100	-	50	150
CSE806-P	Project-2	-	-	2	1	-	120	-	80	200
CSE807-P	Seminar	-	-	2	1	-	60	-	40	100
CSE 808	Discipline & Extra Curricular Activity									50
	Total	**12**	**4**	**08**	**20**					**1000**

IA- Internal Assessment **T- Tutorial**

L- Lecture **P- Practical**

Th- Theory

CSE301ELECTRONIC DEVICES & CIRCUITS		
UNIT	**CONTENTS**	**CONTACT HOURS**
I.	Mobility and conductivity, charge densities in a semiconductor, Fermi Dirac distribution,carrier concentrations and fermi levels in semiconductor, Generation and recombination ofcharges, diffusion and continuity equation, Mass action Law, Hall effect. Junction diodes,Diode as a ckt. element, load line concept, clipping and clamping circuits, Voltagemultipliers.	**8**
II.	Transistor characteristics, Current components, Current gains: alpha and beta. Operatingpoint. Hybrid model, h-parameter equivalent circuits. CE, CB and CC configuration. DCand AC analysis of CE,CC and CB amplifiers. Ebers-Moll model. Biasing & stabilizationtechniques. Thermal runaway, Thermal stability.	**8**
III.	SMALL SIGNAL AMPLIFIERS AT LOW FREQUENCY : Analysis of BJT and FET,RC coupled amplifiers. Frequency response, midband gain, gains at low and high frequency.Miller's Theorem. Cascading Transistor amplifiers, Emitter follower. JFET, MOSFET,Equivalent circuits and biasing of JFET's & MOSFET's. Low frequency CS and CD JFETamplifiers. FET as a voltage variable resistor. Source follower.	**8**
IV.	FEEDBACK AMPLIFIERS : Classification, Feedback concept, Transfer gain withfeedback, General characteristics of negative feedback amplifiers. Analysis of voltageseries,voltage-shunt, current- series and current-shunt feedback amplifier. Stabilitycriterion.	**8**
V.	OSCILLATORS : Classification. Criterion for oscillation. Tuned collector, Hartley,Colpitts, RC Phase shift, Wien bridge and crystal oscillators, Astable, monostable andbistable multivibrators. Schmitt trigger.	**8**
TextBooks/Reference:		
• Electronic devices & circuits theory By R.L. Boylestad, Louis Nashelsky ,Pearson education • Integrated Electronics By Millman Halkias, T.M.H • Electronic devices & circuits By David Bell, Oxford Publications • Grob's Basic Electronics By Schultz, T.M.H.		
CSE301-P ELECTRONIC DEVICES LAB		
1	Plot V-I characteristic of P-N junction diode & calculate cut-in voltage, reverse Saturation current and static & dynamic resistances.	
2	Plot V-I characteristic of P-N junction diode & calculate cut-in voltage, reverse Saturation current and static & dynamic resistances.	
3	Plot frequency response curve for single stage amplifier and to determine gain bandwidth product.	
4	Plot drain current - drain voltage and drain current – gate bias characteristics of field effect transistor and measure of Idss & Vp	
5	Application of Diode as clipper & clamper	
6	Plot gain- frequency characteristic of two stages RC coupled amplifier & calculate its bandwidth and compare it with theoretical value.	
7	Plot gain- frequency characteristic of emitter follower & find out its input and output resistances.	
8	Plot input and output characteristics of BJT in CB, CC and CE configurations. Find their h-parameters.	
9	Plot gain-frequency characteristics of BJT amplifier with and without negative feedback in the emitter circuit and determine bandwidths, gain bandwidth products and gains at 1kHz with and without negative feedback.	
10	Plot and study the characteristics of small signal amplifier using FET.	
11	Plot and study the characteristics of small signal amplifier using FET.	
12	Study transistor phase shift oscillator and observe the effect of variation in R & C on oscillator frequency and compare with theoretical value.	
13	To plot the characteristics of UJT and UJT as relaxation.	
14	To plot the characteristics of MOSFET and CMOS.	

CSE302DATA STRUCTURES & ALGORITHMS

UNIT	CONTENTS	CONTACT HOURS
I.	Definition & characteristics of algorithms, structures. Difficulties in estimatingexact execution time of algorithms. Concept of complexity of program.Asymptotic notations: Big-Oh, theta, Omega- Definitions and examples,Determination of time and space complexity of simple algorithms withoutrecursion. Representing a function in asymptotic notations viz 5n2-6n=_(n2)Arrays: Array as storage element, Row major & column major form of arrays,computation of address of elements of n dimensional array.	8
II.	Arrays as storage elements for representing polynomial of one or more degreesfor addition & multiplication, sparse matrices for transposing & multiplication,stack, queue, dequeue, circular queue for insertion and deletion with conditionfor over and underflow, transposition of sparse matrices with algorithms ofvarying complexity (Includes algorithms for operations as mentioned).Evaluation of Expression: Concept of precedence and associativity inexpressions, difficulties in dealing with infix expressions, Resolving precedence of operators and association of operands, postfix & prefix expressions,conversion of expression from one form to other form using stack (with &without parenthesis), Evaluation of expression in infix, postfix & prefix forms using stack. Recursion.	8
III.	Linear linked lists: singly, doubly and circularly connected linear linked listsinsertion,deletion at/ from beginning and any point in ordered or unordered lists.Comparison of arrays and linked lists as data structures. Linked implementation of stack, queue and dequeue. Algorithms for of insertion,deletion and traversal of stack, queue, dequeue implemented using linkedstructures. Polynomial representation using linked lists for addition, Concepts ofHead Node in linked lists.Searching: Sequential and binary search	8
IV.	Non-Linear Structures: Trees definition, characteristics concept of child, sibling,parent child relationship etc, binary tree: different types of binary trees based ondistribution of nodes, binary tree (threaded and unthreaded) as data structure,insertion, deletion and traversal of binary trees, constructing binary tree from traversal results. Threaded binary Tree. Time complexity of insertion, deletionand traversal in threaded and ordinary binary trees. AVL tree: Concept ofbalanced trees, balance factor in AVL trees, insertion into and deletion fromAVL tree, balancing AVL tree after insertion and deletion. Application of treesfor representation of sets.	8
V.	Graphs: Definition, Relation between tree & graph, directed and undirectedgraph, representation of graphs using adjacency matrix and list. Depth first andbreadth first traversal of graphs, finding connected components and spanningtree. Single source single destination shortest path algorithms.Sorting: Insertion, quick, heap, topological and bubble sorting algorithms fordifferent characteristics of input data. Comparison of sorting algorithms in termof time complexity. NOTE: 1. Algorithm for any operation mentioned with a data structure or required toimplement the particular data structure is included in the curriculum.	8

Text Books/Reference:

- An introduction to data structure with applications by Jean-Pual Tremblay, P. G. Sorenson TMH.
- Data Structure in C/C++, Horowitz, Sawhney ,Galgotia
- Data Structure in C/C++, Tanenbaum, Pearson
- Data Structure in C/C++, Weiss, Person

CSE302-P DATA STRUCTURES LAB

1	Write a simple C program on a 32 bit compiler to understand the concept of array storage, size of a word. The program shall be written illustrating the concept of row major and column major storage. Find the address of element and verify it with the theoretical value. Program may be written for arrays upto 4-dimensions.
2	Simulate a stack, queue, circular queue and dequeue using a one dimensional array as storage element. The program should implement the basic addition, deletion and traversal operations.
3	Represent a 2-variable polynomial using array. Use this representation to implementaddition of polynomials.
4	Represent a 2-variable polynomial using array. Use this representation to implement addition of polynomials.
5	Implement singly, doubly and circularly connected linked lists illustrating operations like addition at different locations, deletion from specified locations and traversal.
6	Repeat exercises 2, 3 & 4 with linked structures.
7	Implementation of binary tree with operations like addition, deletion, traversal.
8	Depth first and breadth first traversal of graphs represented using adjacency matrix and list.
9	Implementation of binary search in arrays and on linked Binary Search Tree.
10	Implementation of insertion, quick, heap, topological and bubble sorting algorithms.

CSE303DIGITAL ELECTRONICS

UNIT	CONTENTS	CONTACT HOURS

I.	**Number Systems, Basic Logic Gates & Boolean Algebra:** Binary Arithmetic & Radix representation of different numbers. Sign & magnitude representation, Fixed point representation, complement notation, various codes & arithmetic in different codes & their inter conversion. Features of logic algebra, postulates of Boolean algebra.Theorems of Boolean algebra.Boolean function. Derived logic gates: Exclusive-OR, NAND, NOR gates, their block diagrams and truth tables. Logic diagrams from Boolean expressions and vica-versa. Converting logic diagrams to universal logic. Positive, negative and mixed logic.Logic gate conversion.	**8**
II.	**Digital Logic Gate Characteristics:** TTL logic gate characteristics: Theory & operation of TTL NAND gate circuitry. Open collector TTL. Three state output logic. TTL subfamilies.MOS & CMOS logic families. Realization of logic gates in RTL, DTL, ECL, C-MOS & MOSFET. Interfacing logic families to one another.	**8**
III.	**Minimization Techniques:** Minterm, Maxterm, Karnaugh Map, K map upto 4 variables. Simplification of logic functions with K-map Conversion of truth tables in POS and SOP form. Incomplete specified functions. Variable mapping. Quinn-McKlusky minimization techniques.	**8**
IV.	**Combinational Systems:** Combinational logic circuit design, half and full adder, subtractor. Binary serial and parallel adders. BCD adder. Binary multiplier. Decoder: Binary to Graydecoder, BCD to decimal, BCD to 7-segment decoder. Multiplexer, demultiplexer, encoder. Octal to binary, BCD to excess-3 encoder .Diode switching matrix. Design of logic circuits by multiplexers, encoders, decoders and demultiplexers.	**8**
V.	**Sequential Systems:** Latches, flip-flops, R-S, D, J-K, Master Slave flip flops. Conversions of flip-flops. **Counters:** Synchronous & asynchronous ripple and decade counters, Modulus counter, skipping state counter, counter design, state diagrams and state reduction techniques. Ring counter. Counter applications. Registers: buffer register, shift register.	**8**

Text Books/Reference:

- Digital integrated electronics, By Herbert Taub, Donald L. Schilling, TMH
- Digital Logic and Computer Design By M. Morris Mano, Pearson
- Modern Digital Electronics By R.P. Jain, TMH
- Fundamentals of Digital circuits By A. Anand kumar, PHI
- Digital circuit design By S. Salivahanan, Sarivazhagan, Vikas publications

CSE303-P DIGITAL ELECTRONICS LAB

1	To verify the truth tables of basic logic gates: AND, OR, NOR, NAND, NOR. Also to verify the truth table of Ex-OR, Ex-NOR (For 2, 3, & 4 inputs using gates with 2, 3, & 4 inputs).
2	To verify the truth table of OR, AND, NOR, Ex-OR, Ex-NOR realized using NAND & NOR gates.
3	To realize an SOP and POS expression.
4	To realize Half adder/ Subtractor & Full Adder/ Subtractor using NAND & NOR gates and to verify their truth tables.
5	To realize a 4-bit ripple adder/ Subtractor using basic Half adder/ Subtractor & basic Full Adder/ Subtractor.
6	To verify the truth table of 4-to-1 multiplexer and 1-to-4 demultiplexer. Realize the multiplexer using basic gates only. Also to construct and 8-to-1 multiplexer and 1-to-8 demultiplexer using blocks of 4-to-1 multiplexer and 1-to-4 demultiplexer
7	Design & Realize a combinational circuit that will accept a 2421 BCD code and drive a TIL -312 seven-segment display.
8	Using basic logic gates, realize the R-S, J-K and D-flip flops with and without clock signal and verify their truth table
9	Construct a divide by 2,4& 8 asynchronous counter. Construct a 4-bit binary counter and ring counter for a particular output pattern using D flip flop.
10	Perform input/output operations on parallel in/Parallel out and Serial in/Serial out registers using clock. Also exercise loading only one of multiple values into the register using multiplexer. Note: As far as possible, the experiments shall be performed on bread board. However, experiment Nos. 1-4 are to be performed on bread board only.

CSE304 LINUX AND SHELL PROGRAMMING

UNIT	CONTENTS	CONTACT HOURS
I.	Introduction: Logging in, changing password (passwd command only), man, xman, info commands to access on line help. Simple commands like ls, cp, mv, grep, head, tail, sort, uniq, diff, echo, date, which, whereis, whatis, who, finger w (option and variations included). Directory commands, access permissions, changing access permissions for files and directories, hard & symbolic links. Environment and path setting.	**8**
II.	vi editor: Creating and editing files, features of vi, insertion deletion, searching, substitution operations, yank, put, delete commands, reading & writing files, exrc file for setting parameters, advance editing techniques. vim(improved vi). Programming utilities: Compiling & linking C, C++ programs, make utility, debugging C programs using gdb, system call.	**8**
III.	Introduction to X-window system: x-window as client/ server system, concept of window manager, remote computing & local displays, xinitrc file, customize X work environment and applications, customizing the fvwm window manager.	**8**
IV.	Shell: Meaning and purpose of shell, Introduction to types of shell. The command line, standard input and standard output, redirection, pipes, filters special characters for searching files and pathnames. Bourne Again SHell: shell script-writing and executing, command separation & grouping, redirection, directory stack manipulation, processes, parameters & variables, keyword variables.	**8**
V.	Shell Programming: Control structures, the Here document, expanding NULL or USET variables, Builtins, functions, history, aliases, job control, filename substitution. source code management- RCS and CVS. awk utility.	**8**

TextBooks/Reference:

- A Practical Guide to Linux, Sobell, Pearson.
- A Practical Guide to Linux Command, Editors, and Shell Programming, Sobell, Pearson.
- A Practical Guide to Fedora & Red Hat Enterprise Linux, Sobell, 5e, Pearson
- Harley Hann: Guide Unix & Linux, TMH

CSE304-P UNIX SHELL PROGRAMMING

1	Use of Basic Unix Shell Commands: ls, mkdir, rmdir, cd, cat, banner, touch, file, wc, sort, cut, grep, dd, dfspace, du, ulimit.
2	Commands related to inode, I/O redirection and piping, process control commands, mails.
3	Shell Programming: Shell script exercises based on following (i) Interactive shell scripts (ii) Positional parameters (iii) Arithmetic (iv) if-then-fi, if-then-else-fi, nested if-else (v) Logical operators (vi) else + if equals elif, case structure (vii) while, until, for loops, use of break (viii) Metacharacters (ix) System administration: disk management and daily administration
4	Write a shell script to create a file in $USER /class/batch directory. Follow the instructions (i) Input a page profile to yourself, copy it into other existing file; (ii) Start printing file at certain line (iii) Print all the difference between two file, copy the two files at $USER/CSC/2007 directory. (iv) Print lines matching certain word pattern.
5	Write shell script for- (i) Showing the count of users logged in, (ii) Printing Column list of files in your home directory (iii) Listing your job with below normal priority (iv) Continue running your job after logging out.
6	Write a shell script to change data format .Show the time taken in execution of this script
7	Write a shell script to print files names in a directory showing date of creation & serial number of the file.
8	Write a shell script to count lines, words and characters in its input(do not use wc).
9	Write a shell script to print end of a Glossary file in reverse order using Array. (Use awk tail)
10	Write a shell script to check whether Ram logged in, Continue checking further after every 30 seconds till success.

CSE305 OBJECT ORIENTED PROGRAMMING

UNIT	CONTENTS	CONTACT HOURS
I.	Introduction: Review of structures in C, accessing members of structures using structure variables, pointer to structures, passing structures to functions, structures as user defined data types.	**8**
II.	Introduction to programming paradigms- (Process oriented and Object oriented). Concept of object, class, objects as variables of class data type, difference in structures and class in terms of access to members, private and public Basics of C++: Structure of C++ programs, introduction to defining member functions within and outside a class, keyword using, declaring class, creating objects, constructors & destructor functions, Initializing member values with and without use of constructors, simple programs to access & manipulate data members, cin and cout functions. Dangers of returning reference to a private data member, constant objects and members function, composition of classes, friend functions and classes, using this pointer, creating and destroying objects dynamically using new and delete operators. Static class members, container classes and iterators, proxy classes. Members of a class, data & function members. Characteristics of OOP- Data hiding, Encapsulation, data security.	**8**
III.	Operator overloading: Fundamentals, Restrictions, operator functions as class members v/s as friend functions. Overloading stream function, binary operators and unary operators. Converting between types.	**8**
IV.	Inheritance: Base classes and derived classes, protected members, relationship between base class and derived classes, constructors and destructors in derived classes, public, private and protected inheritance, relationship among objects in an inheritance hierarchy, abstract classes, virtual functions and dynamic binding, virtual destructors.	**8**
V.	Multiple inheritance, virtual base classes, pointers to classes and class members, multiple class members. Templates, exception handling.	**8**

Text Books/Reference:

- How to program C++,Dietel, Pearson
- Mastering C++By K.R. Venugopal, TMH
- Object Oriented programming in C++ By Robert, Lafore, Pearson
- Object Oriented Design & Modelling, Rambaugh, Pearson

CSE305-P C++ PROGRAMMING LAB

1	To write a simple program for understanding of C++ program structure without any CLASS declaration. Program may be based on simple input output, understanding of keyword using.
2	Write a C++ program to demonstrate concept of declaration of class with public & private member, constructors, object creation using constructors, access restrictions, defining member functions within and outside a class. Scope resolution operators, accessing an object's data members and functions through different type of object handle name of object, reference to object, pointer to object, assigning class objects to each other.
3	Program involving multiple classes (without inheritance) to accomplish a task. Demonstrate composition of class.
4	Demonstration Friend function friend classes and this pointer.
5	Demonstration dynamic memory management using new & delete & static class members.
6	Demonstration of restrictions an operator overloading, operator functions as member function and/ or friend function, overloading stream insertion and stream extraction, operators, overloading operators etc.
7	Demonstrator use of protected members, public & private protected classes, multi-level inheritance etc.
8	Demonstrating multiple inheritance, virtual functions, virtual base classes, abstract classes
9	To write a simple program for understanding of C++ program structure without any CLASS declaration. Program may be based on simple input output, understanding of keyword using.
10	Write a C++ program to demonstrate concept of declaration of class with public & private member, constructors, object creation using constructors, access restrictions, defining member functions within and outside a class. Scope resolution operators, accessing an object's data members and functions through different type of object handle name of object, reference to object, pointer to object, assigning class objects to each other.

CSE306 ADVANCED ENGINEERING MATHEMATICS

UNIT	CONTENTS	CONTACT HOURS
I.	Introduction: Engineering application of optimization, Statement and classification of optimization problem, single variable and multivariable optimization with and without constraints.	**8**
II.	Linear Programming: Formulation of Linear Programming problem, Graphical Approach, General Linear Programming problem, Simplex Method. Duality in Linear Programming and Transportation Problems.	**8**
III.	Elements of Number Theory: Divisibility and Euclid Algorithm, Primes and the Sieve of Eratosthenes, testing for primes, Prime Number Theorem, Euler's, Fermat's Little theorems, Congruences, Computing Inverse in Congruences, Legendre and Jacobi Symbols, Chinese Remainder Theorem, Algebraic Structures in Computing (Definitions, properties and Elementary Operations Only): Groups, subgroup, order of group, cyclic group, ring, field, division algorithm, polynomial over a field. Galois Field	**8**
IV.	LAPLACE TRANSFORM: Laplace transform with its simple properties. Inverse Laplace transform, convolution theorem (without proof), solution of ordinary differential equation with constant coefficient, solution of partial differential equation having constant coefficient with special reference to diffusion, Heat conduction and wave equation. Boundary value problems	**8**
V.	NUMERICAL ANALYSIS: Difference operators forward, backward, central, shift and average operators and relation between them. Newton's and Gauss forward and backward interpolation formula for equal interval, Stirling's formula for central difference. Lagrange's Interpolation formula and Inverse Interpolation. Numerical differentiation by Newton's, Gauss and Sterling's formula. Numerical Integration by Simpson's one third and there eight rule. Numerical Integration of ordinary differential equation of first order by Picard's method, Euler's and modified Euler's method, Milne's method and Runga-Kutta fourth order method. Solution of difference equation.	**8**

Text Books/Reference:

- Elementary of theory with applications: Thomas koshy, 2nd Ed.,Elsevier.
- Operation research By Kanti Swaroop, P.K Gupta & Manmohan, Sultan Chand & Sons.
- Integral Transform By Dr. R.K. Gupta, A.R. Vishishtha, Krishna Prakashan Mandir Meerut.
- Calculus of finite differences & Numerical Analysis By Dr. Gupta & Malik, Krishna Prakashan Mandir Meerut.

CSE401 MICROPROCESSOR AND INTERFACES

UNIT	CONTENTS	CONTACT HOURS
I.	Introduction to Microprocessors, microcontroller; 8085 Microprocessor Architecture, pin description, Bus concept and organization; concept of multiplexing and demultiplexing of buses; concept of static and dynamic RAM, type of ROM, memory map.	**8**
II.	Software architecture registers and signals, Classification of instruction, Instruction set, addressing modes, Assembly Language Programming and Debugging, Programming Technique, instruction Format and timing.	**8**
III.	Advance Assembly Language Programming, Counter and time delay; types of Interrupt and their uses, RST instructions and their uses, 8259 programmable interrupt controller; Macros, subroutine; Stack-implementation and uses with examples; Memory interfacing.	**8**
IV.	8085 Microprocessor interfacing:, 8255 Programmable Peripheral Interface, 8254 programmable interval timer, interfacing of Input/output device, 8279 Key board/Display interface.	**8**
V.	Microprocessor Application: Interfacing scanned multiplexed display and liquid crystal display, Interfacing and Matrix Keyboard, MPU Design; USART 8251, RS232C and RS422A, Parallel interface- Centronics and IEEE 488 .	**8**

Text Books/Reference:

- Microprocessor Architecture, Programming, and Applications with the 8085 By Ramesh S. Gaonkar
- Introduction to Microprocessor By Aditya P. Mathur, TMH
- Microprocessor & Interfacing By Douglas V. Hall, TMH
- Microprocessor & Peripheral By A.K. Ray, K. M. Bhurchandi, TMH

CSE401- P MICROPROCESSOR LAB

1	Add the contents of memory locations XX00 &XX01 & place the result in memorylocation XX02.
2	Add the 16 bit numbers stored in memory location & store the result in another memory location.
3	Transfer a block of data from memory location XX00 to another memory location XX00 in forward & reverse order.
4	Write a program to Swap two blocks of data stored in memory.
5	Write a program to find the square of a number.
6	Write a main program & a conversion subroutine to convert Binary to its equivalent BCD.
7	Write a program to find largest & smallest number from a given array.
8	Write a program to Sort an array in ascending & descending order.
9	Write a program to multiply two 8 bit numbers whose result is 16 bit.
10	Write a program of division of two 8 bit numbers.
11	Generate square wave from SOD pin of 8085 & observe on CRO.
12	Write a program to perform traffic light control operation.
13	Write a program to control the speed of a motor.

CSE402 DISCRETE MATHEMATICAL STRUCTURES

UNIT	CONTENTS	CONTACT HOURS
I.	Sets: Definition and types, Set operations, Partition of set, Cardinality (Inclusion- Exclusion & Addition Principles), Recursive definition of set. Functions: Concept, Some Special Functions (Polynomial, Exponential & Logarithmic, Abslute Value, Floor & Ceiling, Mod & Div Functions), Properties of Functions, Cardinality of Infinite Set, Countable & Uncountable Sets, The Pigeonhole & Generalized Pigeonhole Principles, Composition of Functions.	**8**
II.	Relations: Boolean Matrices, Binary Relation, Adjacency Matrix of Relation, Properties of Relations, Operations on Relations, The Connectivity Relations, Transitive Closure- Warshall's Algorithm, Equivalence relations- Congruence Relations, Equivalence Class, Number of Partitions of a Finite Set, Partial & Total Orderings.	**8**
III.	Proof Methods: Vacuous, Trivial, Direct, Indirect by Contrapositive and Contradiction, Constructive & Non-constructive proof, Counter example. The Division Algorithm, Divisibilty Properties (Prime Numbers & Composite Numbers), Principle of Mathematical Induction, The Second Principle of Mathematical Induction, Fundamental Theorem of Arithmetic. Algorithm Correctness: Partial Correctness, Loop Invariant. Testing the partial correctness of linear & binary search, bubble & selection sorting.	**8**
IV.	Graph Theory: Graphs – Directed, Undirected, Simple,. Adjacency & Incidence, Degre of Vertex, Subgraph, Complete graph, Cycle & Wheel Graph, Bipartite & Complete Bipartite Graph, Weighed Graph, Union of Simple Graphs. Complete Graphs. Isomorphic Graphs, Path, Cycles & Circuits Euclerian & Hamiltonian Graphs. Planar Graph: Kuratowski's Two Graphs, Euler's Formula, Kuratowski's Theorem. Trees: Spanning trees- Kruskal's Algo, Finding Spanning Tree using Depth First Search, Breadth First Search, Complexity of Graph, Minimal Spanning Tree.	**8**
V.	Language of Logic: Proposition, Compound Proposition, Conjunction, Disjunction, Implication, Converse, Inverse & Contrpositive, Biconditional Statements, tautology, Contradiction & Contingency, Logical Equivalences, Quantifiers, Arguments.	**8**

Text Books/Reference:

- Discrete Mathematics with Applications, Koshy, ELSEVIER
- Discrete Mathematical Structures By Lipschutz & Lipson,TMH
- Discrete Mathematical Structures By Kolman et.al, Pearson

CSE403 STATISTICS & PROBABILITY THEORY

UNIT	CONTENTS	CONTACT HOURS
I.	Introduction & Discrete random variables Sample space, events, algebra of events, Bernoulli's trials, Probability& Baye'stheorem. Random variable & their event space, probability generating function, expectations, moments, computations of mean time to failure, Bernoulli & Poissonprocesses.	**8**
II.	Discrete & continuous distributions Probability distribution & probability densities: Binomial, Poisson, normal rectangular and exponential distribution & their PDF's, moments and MGF's for above distributions.	**8**
III.	Correlation & Regression Correlation & regression: Linear regression, Rank correlation, Method of least squares Fitting of straight lines & second degree parabola. Linear regression and correlation analysis.	**8**
IV.	Queuing Theory Pure birth, pure death and birth-death processes. Mathematical models for M/M/1, M/M/N, M/M/S and M/M/S/N queues.	**8**
V.	Discrete Parameter Markov chains: M/G/1 Queuing model, Discrete parameter birth-death process.	**8**

Text Books/Reference:

- Probability, Statistics & Random Process By T. Veerajan, TMH
- Fundamental of Mathematical Statistics By S.C.Gupta and V.K. Kapoor,Sultanchand& sons.
- Statistics and Probability Theory By Jain & Rawat ,CBC
- Statistics and Probability Theory By Schaum's, T.M.H.H. S. Kalsi, Electronic Inst. & Measurement, TMH 2004

CSE 404 Digital Logic Design

UNIT	CONTENTS	CONTACT HOURS
I.	Hardware Description Languages and their use in digital logic design.VHDL: Modelling Concepts, Lexical Elements & Syntax Descriptions, ScalarData types & Operations, Sequential Statements, Composite Data Types &Operations, Basic Modelling Constructs.Case Study: VHDL Simulation of Ripple Carry, & Look Ahead carry Adders.	**8**
II.	VHDL: Subprograms, Packages & Use Clauses, Aliases, Resolved Signals,Components & Configurations, Generate Statements, Concurrent Statements.Use of VHDL in simulation and synthesis.	**8**
III.	Clocked Sequential circuits. Design steps for synchronous sequential circuits.Design of a sequence detector. Moore and Mealy Machines. Design using JKflip-flops and D flip-flops. State reduction, State assignment, Algorithmic StateCharts, converting ASM charts to hardware, one-hot state assignment.Considerations of clock skew, set-up time, hold-time and other flip-flopparameters, timing constraints.Programmable Logic Devices. Read-only memory. Boolean functionimplementation through ROM. PLD, PGA, PLA, PAL, FPGA.	**8**
IV.	Event-driven Circuits. Design procedure for asynchronous circuits, stable andunstable states, races, race-free assignments. State reduction of incompletelyspecified machines. Compatibility and state reduction procedure. Hazards incombinational networks. Dynamic hazards, Function Hazards, and EssentialHazards. Eliminating hazards.	**8**
V.	Field Programmable Gate Arrays: Introduction, Logic Elements &programmability, Interconnect structures & programmability, Extended LogicElements, SRAM, Flash Memory & Antifuse Configuration, Case Studies of Altera Stratix & Xilinx Virtex-II pro.Technology Mapping for FPGAs: Logic Synthesis, Lookup Table Technology Mapping.	**8**

Text Books/Reference:

- Brian Holdsworth and Clive Woods. Digital Logic Design. Newnes (Elsevier). [Available in Indian Edition].
- Ashenden, The Designer's Guide to VHDL, Elsevier.
- Stephen D. Brown, et.al., Field Programmable Gate Arrays, Kluwer Academic Publishers.
- Scott Hauck, André DeHon , Reconfigurable computing: the theory and practice of FPGA based computation, Morgan Kauffman
- Zvi Kohavi: Switching and Finite Automata Theory. TMH.
- Parag K. Lala, Practical Digital Logic Design and Testing. PHI
- Stephen H. Unger, The essence of logic circuits. Wiatrowski & House.

CSE 404-P DIGITAL HARDWARE DESIGN LAB

Objectives: At the end of course, the students shall be able to

1	Should be able to design data path for digital systems
2	Create a digital system using discrete digital ICs
3	Design a hard wired / micro-programmed control circuit
4	Simulate a digital data path in Hardware Description Language
	Understand IC descriptions and select proper IC in a given circuit based on its timing characteristics
	Suggested Methodology and tools: Hardware description language like Verilog/VHDL can be used for simulation. The exercise shall involve design of data path, its simulation and finally realization onbreadboard. Library of digital ICs have to be built. Similarly, manuals of Digital ICfamilies have to be placed in the laboratories for reference by students. **Suggested Exercises** • Create a microprocessor from ALU 74181. For this, the students may design a small instruction set and attach necessary registers and suitable control unit to realize a microprocessor. • Simulate and realize a Cordic calculator. • Simulate & realize a four adder. -Design and simulation of a 4-bit adder -VHDL/Verilog HDL (Hardware description language) -Interfacing 7-segmentdecoder • Combinational Multiplier -4x4-bit multiplier - Binary –to –BCD conversion -Timing Constraints • CRC checksum generator & verifier • Realizing a carry look ahead adder

CSE405 PRINCIPLES OF COMMUNICATION

UNIT	CONTENTS	CONTACT HOURS
I.	ANALOG MODULATION: Concept of frequency translation. Amplitude Modulation:Description of full AM, DSBSC, SSB and VSB in time and frequency domains, methodsof generation & demodulation, frequency division multiplexing (FDM). AngleModulation: Phase and frequency modulation. Descriptions of FM signal in time andfrequency domains, methods of generation & demodulation, pre- emphasis & deemphasis,PLL.	**8**
II.	PULSE ANALOG MODULATION: Ideal sampling, Sampling theorem, aliasing,interpolation, natural and flat top sampling in time and frequency domains. Introductionto PAM, PWM, PPM modulation schemes. Time division multiplexing (TDM)	**8**
III.	PCM & DELTA MODULATION SYSTEMS: Uniform and Non-uniform quantization.PCM and delta modulation, Signal to quantization noise ratio in PCM and deltamodulation. DPCM, ADM, T1 Carrier System, Matched filter detection. Errorprobability in PCM system.	**8**
IV.	DIGITAL MODULATION: Baseband transmission: Line coding (RZ, NRZ), intersymbol interference (ISI), pulse shaping, Nyquist criterion for distortion free base bandtransmission, raised cosine spectrum. Pass band transmission: Geometric interpretationof signals, orthogonalization. ASK, PSK, FSK, QPSK and MSK modulation techniques,coherent detection and calculation of error probabilities.	**8**
V.	SPREAD-SPECTRUM MODULATION: Introduction, Pseudo-Noise sequences, directsequencespread spectrum (DSSS) with coherent BPSK, processing gain, probability oferror, frequency-hop spread spectrum (FHSS). Application of spread spectrum: CDMA.	**8**

Text Books/Reference:

- Principles of communication systems By Taub Schilling, T.M.H.
- Fundamentals of communication systems By Proakis & Salehi, Pearson education
- Communication Systems by Simon Haykin, John Wiley
- Communication Systems (Analog and Digital) By R.P. Singh, S.D. Sapre, T.M.H.
- Modern Digital & Analog Communication By B.P. Lathi, Oxford Publications
- Digital & Analog Communication Systems By K.S. Shanmugam, John Wiley

CSE405- P COMMUNICATION LAB

1	Harmonic analysis of a square wave of modulated waveform Observe the amplitude modulated waveform and measures modulation index. Demodulation of the AM signal
2	To modulate a high frequency carrier with sinusoidal signal to obtain FM signal. Demodulation of the FM signal
3	To observe the following in a transmission line demonstrator kit : i. The propagation of pulse in non-reflecting Transmission line. ii. The effect of losses in Transmission line. iii. The resonance characteristics of al half wavelength long x-mission line.
4	To study and observe the operation of a super heterodyne receiver
5	To modulate a pulse carrier with sinusoidal signal to obtain PWM signal and demodulate it.
6	To modulate a pulse carrier with sinusoidal signal to obtain PPM signal and demodulate it.
7	To observe pulse amplitude modulated waveform and its demodulation.
8	To observe the operation of a PCM encoder and decoder. To consider reason for using digital signal x-missions of analog signals.
9	Produce ASK signals, with and without carrier suppression. Examine the different processes required for demodulation in the two cases
10	To observe the FSK wave forms and demodulate the FSK signals based on the properties of (a) tuned circuits (b) on PI.L..
11	To study & observe the amplitude response of automatic gain controller (AGC).

CSE406 PRINCIPLES OF PROGRAMMING LANGUAGES		
UNIT	**CONTENTS**	**CONTACT HOURS**
I.	Programming Language: Definition, History, Features. Issues in Language Design:Structure and Operation of computer, Programming Paradigms. Efficiency, Regularity.Issues in Language Translation: Syntax and Semantics.	**8**
II.	Specifications and Implementation of Elementary and Structured Data Types. Typeequivalence, checking and conversion. Vectors and Arrays, Lists, Structures, Sets, Files.	**8**
III.	Sequence control with Expressions, Conditional Statements, Loops, Exception handling.Subprogram definition and activation, simple and recursive subprogram, subprogramenvironment.	**8**
IV.	Scope – Static and Dynamic, Block structures, Local Data and Shared Data, Parametersand Parameter Transmission. Local and Common Environments, Tasks and SharedData.	**8**
V.	Abstract Data type, information hiding, encapsulation, type definition. Static and Stack-Based Storage management. Fixed and Variable size heap storage management, GarbageCollection.	**8**
Text Books/Reference:		
• Programming languages: design and implementation, Terrence W. Pratt., Pearson • Programming languages: concepts and constructs, Ravi Sethi, ISBN 9780201590654. • Programming Language Pragmatics, Scott, ELSEVIER		

CSE407- P BUSINESS ENTREPRENEURSHIP DEVELOPMENT	
1	Introduction to Entrepreneurship- Concept and need, Entrepreneurship and innovation,Entrepreneurship and economic growth.
2	Entrepreneurial competencies, Leadership, Decision making, Motivation, Risk taking.
3	Business Enterprise Planning- Identification of business opportunity, Idea generation,Demand estimation, Preparation of project report, Feasibility analysis.
4	Intellectual Property rights, Patents, Taxation- Central excise & Sales tax, VAT.
5	Government Policies for Entrepreneurs, Entrepreneurial career opportunities forEngineers, case studies.

CSE501 COMPUTER ARCHITECTURE

UNIT	CONTENTS	CONTACT HOURS
I.	Introduction to Computer Architecture and Organization. Von NeumanArchitecture, Flynn Classification. Register Transfer and Micro operations: Register transfer language, ArithmeticMicro-operations, Logic Micro-operations, Shift Micro-operations, Bus andmemory transfers.Computer Organization and Design: Instruction cycle, computer registers,common bus system, computer instructions, addressing modes, design of a basic computer	**8**
II.	Central Processing Unit: General register organization, stack organization,Instruction formats, Data transfer and manipulation, program control.RISC, CISC characteristics.Pipeline and Vector processing: Pipeline structure, speedup, efficiency,throughput and bottlenecks. Arithmetic pipeline and Instruction pipeline.	**8**
III.	Computer Arithmetic: Adder, Ripple carry Adder, carry look Ahead Adder,Multiplication: Add and Shift, Array multiplier and Booth Multiplier, Division:restoring and Non-restoring Techniques. Floating Point Arithmetic: Floatingpoint representation, Add, Subtract, Multiplication, Division.	**8**
IV.	Memory Organization: RAM, ROM, Memory Hierarchy, Organization,Associative memory, Cache memory, and Virtual memory: Paging andSegmentation.	**8**
V.	Input-Output Organization: Input-Output Interface, Modes of Transfer, PriorityInterrupt, DMA, IOP processor.	**8**

Text Books/Reference:

- Computer Organization and Architecture - William Stallings (Pearson Education Asia)
- Computer Organization and Architecture -John P. Hayes (McGraw -Hill)
- Computer Organization -V. Carl. Hamacher (McGraw-Hill)

CSE 502 SOFTWARE ENGINEERING

UNIT	CONTENTS	CONTACT HOURS
I.	System Analysis: Characteristics, Problems in system Development, System Level project Planning, System Development Life cycle (SDLC), computer system engineering & system analysis, modeling the architecture, system specification.	**8**
II.	Software & its characteristics: Software Development, Process Model, Prescriptive model, The water fall model, Incremental Process Modes, Evolutionary process model, specialized process model.	**8**
III.	Requirement Analysis: Requirement analysis tasks, Analysis principles, Software prototyping and specification data dictionary finite state machine (FSM) models. Structured Analysis: Data and control flow diagrams, control and process specification behavioral modeling, extension for data intensive applications.	**8**
IV.	Software Design: Design fundamentals, Effective modular design: Data architectural and procedural design, design documentation, coding – Programming style, Program quality,quantifying program quality, complete programming example	**8**
V.	Object Oriented Analysis: Object oriented Analysis Modeling, Data modelling Object Oriented Design: OOD concepts and methods class and object definitions, refining operations, Class and object relationships, object modularization, Introduction to Unified Modeling Language	**8**

Text Books/Reference:

- Software Engineering By Roger S. Pressman, TMH
- Software Engineering Fundamental By Ali Behforooz, Frederick J Hudson, Oxford University Press
- Software Engineering By Ian Sommerville
- Software Engineering Concepts By Richard E. Fairley (Mcgraw-Hill)

CSE 502-P COMPUTER AIDED SOFTWARE ENGINEERING LAB

For the instructor: Assign any two projects two a group of exactly two students covering all of the experiments from given experiment list. Each group is required to prepare the following documents for projects assigned to them and develop the software using software engineering methodology.

1	Problem Analysis and Project Planning Thorough study of the problem- identify project scope, infrastructure.
2	Software Requirement Analysis- Describe the individual Phases/modules of the project deliverables.
3	Distribution system Design: Design of feeders & distributors. Calculation of voltage drops in distributors. Calculation of conductor size using Kelvin's law.
4	Software Developments and Debugging.
5	Software Testing – Prepare test plan, perform validation testing coverage analysis, memory leaks, develop test case hierarchy, Site check and site monitor.
6	Describe: Relevance of CASE tools for data dictionaries

S. No.	List of Experiments	Software Recommended:
1	Course Registration System	Case Tools: Rational Suite, Win runner, Empirix Languages: C/C++/JDK, JSDK, INTERNET EXPLORER UML Front End: VB, VC++, Developer 2000, .NET Back End: Oracle, MS – Access, SQL Note: Open Source tools will be preferred.
2	Quiz System	
3	Online ticket reservation system	
4	Remote computer monitoring	
5	Students marks analyzing system	
6	Expert system to prescribe the medicines for the given symptoms	
7	Platform assignment system for the trains in arailway station	
8	Stock maintenance	
9	Student Marks Analyzing System	
10	Online Ticket Reservation System	
11	Payroll System	
12	Export System	

CSE 503 THEORY OF COMPUTATION

UNIT	CONTENTS	CONTACT HOURS
I.	Finite Automata & Regular Expression: Basic Concepts of finite state system, Deterministic and non-deterministic finite automation and designing regular expressions, relationship between regular expression & Finite automata minimization of finite automation mealy & Moore Machines.	**8**
II.	Regular Sets of Regular Grammars: Basic Definition of Formal Language and Grammars. Regular Sets and Regular Grammars, closure proportion of regular sets, Pumping lemma for regular sets, decision Algorithms for regular sets, Myhell_Nerod Theory & Organization of Finite Automata.	**8**
III.	Context Free Languages& Pushdown Automata: Context Free Grammars – Derivations and Languages – Relationship between derivation and derivation trees – ambiguity – simplification of CEG – Greiback Normal form – Chomsky normal forms – Problems related to CNF and GNF Pushdown Automata: Definitions – Moves – Instantaneous descriptions – Deterministic pushdown automata – Pushdown automata and CFL - pumping lemma for CFL - Applications of pumping Lemma.	**8**
IV.	Turing Machines: Turing machines – Computable Languages and functions – Turing Machine constructions – Storage in finite control – multiple tracks – checking of symbols – subroutines – two way infinite tape. Undecidability: Properties of recursive and Recursively enumerable languages – Universal Turing Machines as an undecidable problem – Universal Languages – Rice's Theorems.	**8**
V.	Linear bounded Automata Context Sensitive Language: Chomsky Hierarchy of Languages and automata, Basic Definition &descriptions of Theory & Organization of Linear bounded Automata Properties of context-sensitive languages	**8**

Text Books/Reference:

- Aho, Hopcroft and Ullman, Introduction to Automata Theory, Formal Languages and Computation, Narosa
- Cohen, Introduction to Computer Theory, Addison Wesley.
- Papadimitriou, Introduction to Theory of Computing, Prentice Hall.

CSE504 DATABASE MANAGEMENT SYSTEMS

UNIT	CONTENTS	CONTACT HOURS
I.	INTRODUCTION TO DATABASE SYSTEMS: Overview and History of DBMS.File System v/s DBMS .Advantage of DBMS Describing and Storing Data in aDBMS. Queries in DBMS. Structure of a DBMS.	**8**
II.	ENTITY RELATIONSHIP MODEL: Overview of Data Design Entities, Attributesand Entity Sets, Relationship and Relationship Sets. Features of the ER Model- KeyConstraints, Participation Constraints, Weak Entities, Class Hierarchies, Aggregation,Conceptual Data Base, Design with ER Model-Entity v/s Attribute, Entity vs Relationship Binary vs Ternary Relationship and Aggregation v/s ternary RelationshipConceptual Design for a Large Enterprise.	**8**
III.	RELATIONSHIP ALGEBRA AND CALCULUS**:** Relationship Algebra Selectionand Projection, Set Operations, Renaming, Joints, Division, Relation Calculus,Expressive Power of Algebra and Calculus.	**8**
IV.	SQL QUERIES PROGRAMMING AND TRIGGERS: The Forms of a Basic SQLQuery, Union, Intersection and Except, Nested Queries ,Correlated Nested Queries,Set-Comparison Operations, Aggregate Operators, Null Values and Embedded SQL,Dynamic SQL, ODBC and JDBC, Triggers and Active Databases.	**8**
V.	SCHEMA REFINEMENT AND NORMAL FORMS: Introductions to SchemaRefinement, Functional Dependencies, Boyce-Codd Normal Forms, Third NormalForm, Normalization-Decomposition into BCNF Decomposition into 3-NF.	**8**

Text Books/Reference:

- H.f. Korth and Silberschatz: Database Systems Concepts, McGraw Hill
- Almasri and S.B. Navathe: Fundamentals of Database Systems,
- C.J. Date: Data Base Design, Addison Wesley
- Hansen and Hansen : DBM and Design, PHI

CSE504-P DATABASE LAB

Objectives: At the end of the semester, the students should haveclearly understood and implemented the following:

1	Stating a database design & application problem.
2	Preparing ER diagram
3	Finding the data fields to be used in the database.
4	Selecting fields for keys.
5	Normalizing the database including analysis of functionaldependencies.
6	Installing and configuring the database server and the front end tools.
7	Designing database and writing applications for manipulation of data fora standalone and shared data base including concepts like concurrencycontrol, transaction roll back, logging, report generation etc.
8	**Get acquainted with SQL.** In order to achieve the above objectives, it is expected that each students willchose one problem. The implementation shall being with the statement of theobjectives to be achieved, preparing ER diagram, designing of database,normalization and finally manipulation of the database including generation ofreports, views etc. The problem may first be implemented for a standalonesystem to be used by a single user. All the above steps may then be followed for development of a databaseapplication to be used by multiple users in a client server environment withaccess control. The application shall NOT use web techniques. One exercise may be assigned on creation of table, manipulation of data andreport generation using SQL. **Suggested Tool:** For standalone environment, Visual FoxPro or any similar database having both the database and manipulation language may be used. For multi-user application, MYSql is suggested. However, any other databasemay also be used. For front end, VB.Net, Java, VB Script or any otherconvenient but currently used by industry may be chosen. **Indicative List of exercise:** 1. Student information system for your college. 2. Student grievance registration and redressal system. 3. A video library management system for a shop. 4. Inventory management system for a hardware/ sanitary item shop. 5. Inventory management system for your college. 6. Guarantee management system for the equipments in your college.

CSE505 OPERATING SYSTEMS

UNIT	CONTENTS	CONTACT HOURS
I.	Introduction and need of operating system, layered architecture/logical structure ofoperating system, Type of OS, operating system as resource manager and virtual machine, OSservices, BIOS, System Calls/Monitor Calls, Firmware- BIOS, Boot Strap Loader.Process management- Process model, creation, termination, states & transitions, hierarchy,context switching, process implementation, process control block, Basic System calls- Linux &Windows. Threads- processes versus threads, threading, concepts, models, kernel & userlevel threads, thread usage, benefits, multithreading models.	8
II.	Interprocess communication- Introduction to message passing, Race condition, criticalsection problem, mutual exclusion with busy waiting- disabling interrupts, lock variables, strictalteration, Peterson's solution, TSL instructions, busy waiting, sleep and wakeup calls,semaphore, monitors, classical IPC problems.Process scheduling- Basic concepts, classification, CPU and I/O bound, CPU schedulershort,medium, long-term, dispatcher, scheduling:- preemptive and non-preemptive, Staticand Dynamic Priority, Co-operative & Non-cooperative, Criteria/Goals/Performance Metrics,scheduling algorithms- FCFS, SJFS, shortest remaining time, Round robin, Priority scheduling,multilevel queue scheduling, multilevel feedback queue scheduling, Fair share scheduling.	8
III.	Deadlock- System model, resource types, deadlock problem, deadlock characterization,methods for deadlock handling, deadlock prevention, deadlock avoidance, deadlock detection,recovery from deadlock. Memory management- concepts, functions, logical and physical address space, addressbinding, degree of multiprogramming, swapping, static & dynamic loading- creating a loadmodule, loading, static & dynamic linking, shared libraries, memory allocation schemesfirstfit, next fit, best fit, worst fit, quick fit. Free space management- bitmap, linklist/free list, buddy's system, memory protection and sharing, relocation and addresstranslation.	8
IV.	Virtual Memory- concept, virtual address space, paging scheme, pure segmentationand segmentation with paging scheme hardware support and implementation details,memory fragmentation, demand paging, pre-paging, working set model, page fault frequency,thrashing, page replacement algorithms- optimal, NRU, FIFO, second chance, LRU, LRUapproximationclock, WS clock; Belady's anomaly, distance string; design issues for paging system- local versus global allocation policies, load control, page size, separateinstruction and data spaces, shared pages, cleaning policy, TLB (translation look asidebuffer) reach, inverted page table, I/O interlock, program structure, page fault handling,Basic idea of MM in Linux & windows.	8
V.	File System- concepts, naming, attributes, operations, types, structure, file organization &access(Sequential, Direct ,Index Sequential) methods, memory mapped files, directorystructures- one level, two level, hierarchical/tree, acyclic graph, general graph, file systemmounting, file sharing, path name, directory operations, overview of file system in Linux &windows.Input/Output subsystems- concepts, functions/goals, input/output devices- block andcharacter, spooling, disk structure & operation, disk attachment, disk storage capacity,disk scheduling algorithm- FCFS, SSTF, scan scheduling, C-scan schedule.	8

Text Books/Reference:

- A. Silberschatz and Peter B Galvin: Operating System Principals, Wiley India Pvt. Ltd.
- Achyut S Godbole: Operating Systems, Tata McGraw Hill
- Tanenbaum: Modern Operating System, Prentice Hall.
- DM Dhamdhere: Operating Systems – A Concepts Based Approach, Tata McGraw Hill
- Charles Crowly: Operating System A Design – Oriented Approach, Tata McGraw Hill

CSE505-P OPERATING SYSTEMS SIMULATION LAB

1	Understand the basic functions of operating systems.
2	In depth knowledge of the algorithms used for implementing the tasks performedby the operating systems.
3	Understand & simulate strategies used in Linux & Windows operating systems.
4	Develop aptitude for carrying out research in the area of operating system.
	Suggested Tools: Operating system simulator- MOSS preferably on Linux platform (Available for freedownload from http://www.ontko.com/moss/). **Recommended Exercises:** A. Exercises shall be given on simulation of algorithms used for the tasksperformed by the operating systems. Following modules of the simulator maybe used: Scheduling Deadlock Memory Management Systems File system simulator Algorithms described in the text may be assigned. The simulation results such asaverage latency, hit & Miss Ratios or other performance parameters may becomputed. B. One exercise shall be on simulation of algorithms reported in the recentConferences/ journals and reproducing the results reported therein.

CSE506-IADVANCED DATA STRUCTURE

UNIT	CONTENTS	CONTACT HOURS
I.	ADVANCED TREES: Definitions, Operations on Weight Balanced Trees(Huffman Trees), 2-3 Trees and Red-Black Trees. Dynamic Order Statistics,Interval Tree; Dictionaries.	8
II.	MERGEABLE HEAPS: Mergeable Heap Operations, Binomial Trees,Implementing Binomial Heaps and its Operations, 2-3-4. Trees and 2-3-4 Heaps.Amortization analysis and Potential Function of Fibonacci Heap, ImplementingFibonacci Heap.	8
III.	GRAPH THEORY DEFINITIONS: Definitions of Isomorphic Components.Circuits, Fundamental Circuits, Cut-sets. Cut- Vertices Planer and Dual graphs,Spanning Trees, Kuratovski's two Graphs.GRAPH THEORY ALGORITHMS: Algorithms for Connectedness, Finding allSpanning Trees in a Weighted Graph, Breadth First and Depth First Search,Topological Sort, Strongly Connected Components and Articulation Point.Single Min-Cut Max-Flow theorem of Network Flows. Ford-Fulkerson MaxFlow Algorithms.	8
IV.	SORTING NETWORK: Comparison network, zero-one principle, bitonic sortingand merging network sorter. Priority Queues and Concatenable Queues using 2-3 Trees.Operations on Disjoint sets and its union-find problem, Implementing Sets.	8
V.	NUMBER THEORITIC ALGORITHM: Number theoretic notions, Divisiontheorem, GCD, recursion, Modular arithmetic, Solving Modular Linear equation,Chinese Remainder Theorem, power of an element, Computation of DiscreteLogarithms, primality Testing and Integer Factorization.	8

Text Books/Reference:

- Cormen, Leiserson, Rivest: Introduction to Algorithms, Prentice Hall of India.
- Horowitz and Sahani: Fundamental of Computer algorithms.
- Aho A.V , J.D Ulman: Design and analysis of Algorithms, Addison Wesley
- Brassard : Fundamental of Algorithmics, PHI.

CSE506-IIDIGITAL SIGNAL PROCESSING

UNIT	CONTENTS	CONTACT HOURS
I.	INTRODUCTION: Discrete time signals and systems, properties of discrete time systems, Linear time invariant systems - discrete time. Properties of LTI systems and their block diagrams. Convolution, Discrete time systems describedby difference equations.	8
II.	Fourier Transform: Discrete time Fourier transform for periodic and aperiodicsignal. Properties of DTFT. Z-transform: The region of convergence for the transform.The Inverse Z-transform. Properties of Z transform.	8
III.	SAMPLING: Mathematical theory of sampling. Sampling theorem. Ideal &Practical sampling. Interpolation technique for the reconstruction of a signal fromits samples. Aliasing. Sampling in freq. domain. Sampling of discrete timesignals.	8
IV.	THE DISCRETE FOURIER TRANSFORMS (DFT): Properties of the DFT,Linear Convolution using DFT. Efficient computation of the DFT: Decimation–in-Time and Decimation-in frequency FFT Algorithms.	8
V.	FILTER DESIGN TECHNIQUES: Structures for discrete-time systems- Blockdiagram and signal flow graph representation of LCCD (LCCD – LinearConstant Coefficient Difference) equations, Basic structures for IIR and FIRsystems, Transposed forms. Introduction to filter Design: Butterworth &Chebyshev.IIR filter design by impulse invariance & Bilinear transformation.Design of FIR filters by Windowing: Rectangular, Hamming & Kaiser.	8

Text Books/Reference:

- Oppenheim, Discrete-Time Signal Processing, 2/e, Pearson Education
- Proakis, Digital Signal Processing, 4/e, Pearson Education
- S.K.Mitra, Digital Signal Processing, 2/e, Tata McGraw Hill

CSE506-IIIINFORMATION THEORY & CODING		
UNIT	**CONTENTS**	**CONTACT HOURS**
I.	Introduction to information theory. Uncertainty, Information and Entropy,Information measures for continuous random variables, source coding theorem.Discrete Memory less channels, Mutual information, Conditional entropy.	**8**
II.	Source coding schemes for data compaction: Prefix code, Huffman code,Shanon-Fane code & Hempel-Ziv coding channel capacity. Channel codingtheorem. Shannon limit.	**8**
III.	Linear Block Code: Introduction to error connecting codes, coding & decodingof linear block code, minimum distance consideration, conversion of nonsystematic form of matrices into systematic form.	**8**
IV.	Cyclic Code: Code Algebra, Basic properties of Galois fields (GF) polynomialoperations over Galois fields, generating cyclic code by generating polynomial,parity check polynomial. Encoder & decoder for cyclic codes.	**8**
V.	Convolutional Code: Convolutional encoders of different rates. Code Tree, Trllis and state diagram. Maximum likelihood decoding of convolutional code: The viterbi Algorithm fee distance of a convolutional code.	**8**
Text Books/Reference:		
• Digital Communication, Simon Haykin, Wiley.		

CSE507-P SYSTEM DESIGNS in UML LAB	
Objectives:	
1	The students shall be able to use following modules of UML for systemdescription, implementation and finally for product development. - Capture a business process model. - The User Interaction or Use Case Model - describes the boundary and interaction between the system and users. Corresponds in some respects to a requirements model. - The Interaction or Communication Model - describes how objects in the system will interact with each other to get work Done. - The State or Dynamic Model - State charts describe the states or conditions that classes assume over time. Activity graphs Describe the workflows the system will implement. - The Logical or Class Model - describes the classes and objects that will make up the system. - The Physical Component Model - describes the software (and sometimes hardware components) that make up the System. - The Physical Deployment Model - describes the physical architecture and the deployment of components on that **Hardware architecture.** The students are expected to use the UML models, prepare necessarydocuments using UML and implement a system. Some hardware products likedigital clock, digital camera, washing machine controller, air conditionerController, an electronic fan regulator, an elementary mobile phone etc. mayalso be chosen. The students shall be assigned one problem on software based systems andanother involving software as well as hardware.

CSE601 COMPUTER NETWORKS

UNIT	CONTENTS	CONTACT HOURS
I.	Network layer-design issue, routing algorithms: Distance vector, link state,hierarchical, Broadcast routing. Congestion control: congestion prevention policies, congestion control inDatagram subnets, load shedding, jitter control, Leaky bucket and token bucketalgorithms.	**8**
II.	Internetworking: Differences in networks, Tunneling, Internetwork routing,Fragmentation Network layer in the Internet: IPv4 classful and classlessaddressing, subnetting Network layer protocols(only working and purpose;packet headers etc. not included), Differences in IPV6 over IPV4. Routing toMobile Hosts and Mobile IP	**8**
III.	Elements of transport protocols: addressing, connection establishment andrelease, flow control and buffering, multiplexing and demultiplexing, crashrecovery, introduction to UDP protocol.Principles of Reliable Data Transfer: Reliable data transfer over a perfectlyreliable channel, Channel with bit errors and Lossy Channel with bit errors.	**8**
IV.	Transport Layer in the Internet: Introduction to TCP, TCP service Model, TCPHeader and segment structure, TCP connection establishment and release,transmission policy, timer management, Transactional TCP. Mobile TCPTCP Congestion Control: Fairness, TCP delay modeling.	**8**
V.	Application Layer: World Wide Web (WWW), Domain Name System (DNS),E-mail, File Transfer Protocol (FTP), Introduction to Network security.P2P File Sharing: Centralized Directory, Query flooding, exploitingheterogeneity.	**8**

Text Books/Reference:

- Tanenbaum; Computer Network, 4th Ed., Pearson.
- Kurose; Computer Networking, 3rd Ed., Pearson.
- Peterson, Davie; Computer Networks, 4rd Ed., ELSEVIER

CSE602 DESIGN AND ANALYSIS OF ALGORITHMS

UNIT	CONTENTS	CONTACT HOURS
I.	BACKGROUND: Review of Algorithm Complexity, Order Notations: definitions and calculating complexity. DIVIDE AND CONQUER METHOD: Binary Search, Merge Sort, Quick sort and Strassen's matrix multiplication algorithms. GREEDY METHOD: Knapsack Problem, Job Sequencing, Optimal Merge Patterns and Minimal Spanning Trees.	**8**
II.	DYNAMIC PROGRAMMING: Matrix Chain Multiplication. LongestCommon Subsequence and 0/1 Knapsack Problem. BRANCH AND BOUND: Traveling Salesman Problem and LowerBound Theory. Backtracking Algorithms and queens problem.	**8**
III.	PATTERN MATCHING ALGORITHMS: Naïve and Rabin Karpstring matching algorithms, KMP Matcher and Boyer Moore Algorithms. ASSIGNMENT PROBLEMS: Formulation of Assignment andQuadratic Assignment Problem.	**8**
IV.	RANDOMIZED ALGORITHMS. Las Vegas algorithms, MonteCarlo algorithms, randomized algorithm for Min-Cut, randomized algorithm for 2- SAT. Problem definition of Multicommodity flow, Flow shop scheduling and Network capacity assignment problems.	**8**
V.	PROBLEM CLASSES NP, NP-HARD AND NP-COMPLETE: Definitionsof P,NP-Hard and NP-Complete Problems. Decision Problems. Cook'sTheorem. Proving NP-Complete Problems - Satisfiability problem andVertex Cover Problem. Approximation Algorithms for Vertex Cover andSet Cover Problem. PROBLEM CLASSES NP, NP-HARD AND NPCOMPLETE:Definitions of P, NP-Hard and NP-Complete Problems.Decision Problems. Cook's Theorem. Proving NP-Complete Problems -Satisfiability problem and Vertex Cover Problem. ApproximationAlgorithms for Vertex Cover and Set Cover Problem.	**8**

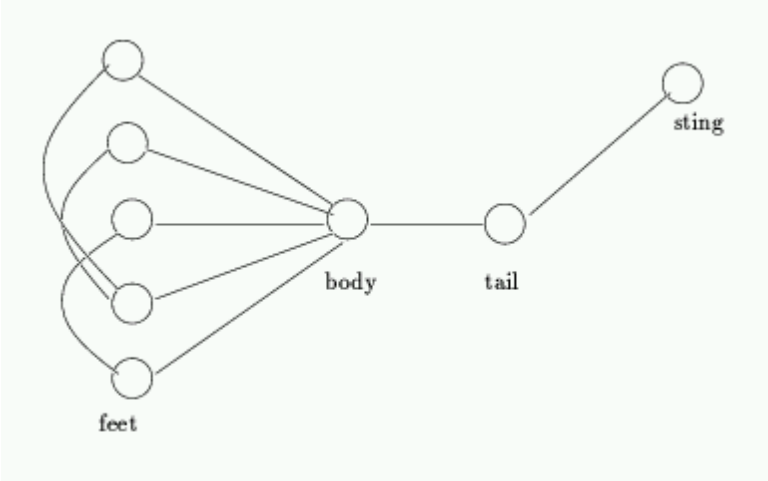
Text Books/Reference:

- Cormen, Leiserson, Rivest: Introduction to Algorithms, Prentice Hall of India.
- Horowitz and Sahani: Fundamental of Computer algorithms.
- Aho A.V , J.D Ulman: Design and analysis of Algorithms, Addison Wesley

CSE602-P DESIGN AND ANALYSIS OF ALGORITHMS Lab

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

	• Prove the correctness and analyze the running time of the basic algorithms for those classic problems in various domains; • Apply the algorithms and design techniques to solve problems; • Analyze the complexities of various problems in different domains. **Suggested Tools:** For implementation and estimation of running time on various sizesof input(s) or output(s) as the case may be, Linux platform is suggested. **Suggested Exercises:** A. It is expected that teachers will assign algorithms to the students for estimation of time & space complexity. Algorithms reported in various research journalsmay be chosen by the teachers. B. Problem on designing algorithms to meet complexity constraints may beassigned. For example, a problem on design, analysis and implementation fortransposing a sparse matrix requiring not more than one pass from the originalmatrix may be assigned. C. A guide to such problems is given below:
1	Exploring a Binary Heap: Consider a binary heap containing n numbers (the root stores the greatest number). You are given a positive integer k < n and a number x. You have to determine whether the kth largest element of the heap is greater than x or not. Your algorithm must take O(k) time. You may use O(k) extra storage.
2	Merging two search trees: You are given two height balanced binary search trees T and T', storing m and n elements respectively. Every element of tree T is smaller than every element of tree T'. Every node u also stores height of the subtree rooted at it. Using this extra information how can you merge the two trees in time O(log m + log n) (preserving both the height balance and the order)?
3	Complete binary tree as an efficient data-structure: You are given an array of size n (n being a power of two). All the entries of the array are initialized to zero. You have to perform a sequence of the following online operations : 1. (i) Add(i,x) which adds x to the entry A[i]. 2. (ii) Report sum(i,j) = sum of the entries in the array from indices I to j for any 0 < i < j <= n.Cohen Sutherland 2D line clipping and Windowing

	It can be seen easily that we can perform the first operation in O(1)time whereas the second operation may cost O(n) in worst case. Yourobjective is to perform these operations efficiently. Give a datastructurewhich will guarantee O(log n) time per operation.
4	Problems on Amortized Analysis a. Delete-min in constant time!!! Consider a binary heap of size n, the root storing the smallest element. We know that the cost of insertion of an element in the heap is O(log n) and the cost of deleting the smallest element is also O(log n). Suggest a valid potential function so that the amortized cost of insertion is O(log n) whereas amortized cost of deleting the smallest element is O(1). b. Implementing a queue by two stack c. Show how to implement a queue with two ordinary stacks so that the amortized cost of each Enqueue and each equeue operation is O(1).
5	Computing a spanning tree having smallest value of largest edge weight: Describe an efficient algorithm that, given an undirected graph G, determines a spanning tree of G whose largest edge weight is minimum over all spanning trees of G.
6	Shortest Path Problems: i. From a subset of vertices to another subset of vertices a. Given a directed graph G(V,E), where edges have nonnegative weights. S and D are two disjoint subsets of the set of vertices. Give an O(|V| log |V| + |E|) time algorithm to find the shortest path among the set of paths possible from any node in S to any node in D. ii. Paths in Directed Acyclic Graph a. Counting the number of paths Given two nodes u,v in a directed acyclic graph G(V,E). Give an O(|E|) time algorithm to count all the paths from u to v. b. Path passing through a subset of nodes Given two nodes u,v and a set of vertices w1, w2,...,wk in a directed acyclic graph G(V,E). Give an O(|E|) time algorithm to output a path(if exists) from u to v which passes through each of the nodes w1,...,wk. If there is no such path then your algorithm must report that "no such path exists".
7	Searching for a friend: You are standing at a crossing from where there emerge four roads extending to infinity. Your friend is somewhere on one of the four roads. You do not know on which road he is and how far he is from you. You have to walk to your friend and the total distance traveled by you must be at most a constant times the actual distance of your friend from you. In terminology of algorithms, you should traverse O(d) distance, where d is the distance of your friend from you.
8	A simple problem on sorted array: Design an O(n)-time algorithm that, given a real number x and a sorted array S of n numbers, determines whether or not there exist two elements in S whose sum is exactly x .
9	Finding the decimal dominant in linear time: You are given n real numbers in an array. A number in the array is called a decimal dominant if it occurs more than n/10 times in the array. Give an O(n) time algorithm to determine if the given array has a decimal dominant.
10	Finding the first one: You are given an array of infinite length containing zeros followed by ones. How fast can you locate the first one in the array?
11	Searching for the Celebrity: Celebrity is a person whom everybody knows but he knows nobody. You have gone to a party. There are total n persons in the party. Your job is to find the celebrity in the party. You can ask questions of the form Does Mr. X know Mr. Y ?. You will get a binary answer for each such question asked. Find the celebrity by asking only O(n) questions.
12	Checking the Scorpion: An n-vertex graph is a scorpion if it has a vertex of degree 1(the sting) connected to a vertex of degree two (the tail) connected to a vertex of degree n-2 (the body) connected to the other n-3 (the feet). Some of the feet may be connected to other feet. Design an algorithm that decides whether a given adjacency matrix represents a scorpion by examining only O(n) entries. 
13	Endless list: You are having a pointer to the head of singly linked list. The list either terminates at null pointer or it loops back to some previous location(not necessarily to the head of the list). You have to determine whether the list loops back

	or ends at a null location in time proportional to the length of the list. You can use at most a constant amount of extra storage.
14	Nearest Common Ancestor: Given a rooted tree of size n. You receive a series of online queries: "Give nearest common ancestor of u, v ". Your objective is to preprocess the tree in O(n) time to get a data structure of size O(n) so that you can answer any such query in O(log n) time.

CSE 603 TELECOMMUNICATION FUNDAMENTALS

UNIT	CONTENTS	CONTACT HOURS
I.	Data Transmission: Terminology, Frequency, spectrum, bandwidth, analog anddigital transmission, Transmission impairments, channel capacity, TransmissionMedia.Wireless Transmission: Antenna and antenna gain.Network Reference Models (OSI/ISO and TCP/IP)Physical Layer: Line Encoding Schemes. Concept of bit period, effect of clockskew, Synchronous and Asynchronous communication.Data Link Layer: Functions of data link layer and design issuesFlow Control: Flow control in loss less and lossy channels using stop-and-wait,sliding window protocols. Performance of protocols used for flow control.	**8**
II.	Error Control Coding: Error Detection, Two Dimensional Parity Checks, andInternet Checksum. Polynomial Codes, Standardized polynomial codes, errordetecting capability of a polynomial codes. Linear codes, performance of linearcodes, error detection & correction using linear codes.Data Link Control: HDLC & PPP including frame structures.MAC sublayer: Channel Allocation Problem, Pure and slotted Aloha, CSMA, CSMA/CD, collision free multiple access. Throughput analysis of pure and slottedAloha. Ethernet Performance.	**8**
III.	Wireless LAN: Hidden node and Exposed node Problems, RTS/CTS basedprotocol, 802.11 Architecture, protocol stack, Physical layer, MAC Sublayer.Bluetooth Architecture and Protocol StackData Link Layer Switching: Bridges (Transparent, Learning and Spanning Tree),Virtual LANs	**8**
IV.	Multiplexing: Frequency division, time division (Synchronous and statistical)multiplexing. ADSL, DS1 and DS3 carriers.Multiple Accesses: TDMA frame structure, TDMA Burst Structure, TDMA Frameefficiency, TDMA Superframe structure, Frame acquisition and synchronization,Slip rate in digital terrestrial networks. Switching: Qualitative description of Space division, time division and space-timespacedivision switching.	**8**
V.	Spread Spectrum Techniques: Direct sequence(DSSS) & frequencyhopping(FHSS); Performance consideration in DSSS & FHSS; Code divisionMultiple access (CDMA): frequency & channel specifications, forward & reverse CDMA channel, pseudo noise(PN) sequences, m-sequence, gold sequence,orthogonal code, gold sequences, Walsh codes, synchronization, power control,handoff, capacity of CDMA system, IMT-2000, WCDM	**8**

Text Books/Reference:

- Stallings, Data and computer communication, 8th ed. Pearson
- Tri.T.Ha, Digital Satellite Communications, 2/e, Tata McGraw Hill
- Alberto Leon-Garcia, Indra Widjaja, COMMUNICATION NETWORKS, 2nd ed., TMH
- Wireless Communications, 2/e, Rappaport, PHI
- Analysis of Computer and Communication Networks, ISBN: 0387744363, Fayez Gebali, 2008, Springer-verlag, 1st Ed.

CSE604 COMPUTER GRAPHICS & MULTIMEDIA TECHNIQUES.

UNIT	CONTENTS	CONTACT HOURS
I.	Introduction to Raster scan displays, Storage tube displays, refreshing, flicking, interlacing, color monitors, display processors, resolution, Introduction to Interactive. Computer Graphics: Picture analysis, Overview of programmer's model of interactive graphics, Fundamental problems in geometry. Scan Conversion: point, line, circle, ellipse polygon, Aliasing, and introduction to Anti Aliasing (No anti aliasing algorithm).	**8**
II.	2D & 3D Co-ordinate system**:** Homogeneous Co-ordinates, Translation, Rotation, Scaling, Reflection, Inverse transformation, Composite transformation. Polygon Representation, Flood Filling, Boundary filling. Point Clipping, Cohen-Sutherland Line Clipping Algorithm, Polygon Clipping algorithms.	**8**
III.	Hidden Lines & Surfaces: Image and Object space, Depth Buffer Methods, Hidden Facets removal, Scan line algorithm, Area based algorithms. Curves and Splines: Parametric and Non parametric Representations, Bezier curve, Bspline Curves.	**8**
IV.	Rendering: Basic illumination model, diffuse reflection, specular reflection, phong shading, Gourand shading, ray tracing, color models like RGB, YIQ, CMY, HSV	**8**
V.	Multimedia components, Multimedia Input/Output Technologies: Storage and retrieval technologies, Architectural and telecommunication considerations. Animation: Introduction, Rules, problems and Animation techniques.	**8**

Text Books/Reference:

- J. Foley, A. Van Dam, S. Feiner, J. Hughes: Computer Graphics- Principles and Practice, Pearson
- Hearn and Baker: Computer Graphics, PHI
- Multimedia Systems Design, Prabhat Andleigh and Thakkar, PHI.
- Multimedia Information Networking, N.K.Sharda, PHI..

CSE604-P Computer graphics & multimedia lab

1	Implementation of Line, Circle and ellipse attributes
2	Two Dimensional transformations - Translation, Rotation, Scaling, Reflection, Shear
3	Composite 2D Transformations
4	Cohen Sutherland 2D line clipping and Windowing
5	Sutherland – Hodgeman Polygon clipping Algorithm
6	Three dimensional transformations - Translation, Rotation, Scaling
7	Composite 3D transformations
8	Drawing three dimensional objects and Scenes
9	Generating Fractal images
10	To plot a point (pixel) on the screen
11	To draw a straight line using DDA Algorithm
12	Implementation of mid-point circle generating Algorithm
13	Implementation of ellipse generating Algorithm
14	To translate an object with translation parameters in X and Y directions
15	To scale an object with scaling factors along X and Y directions
16	To rotate an object with a certain angle about origin
17	Perform the rotation of an object with certain angle about an arbitrary point

CSE605 EMBEDDED SYSTEM DESIGN

UNIT	CONTENTS	CONTACT HOURS
I.	Introduction to embedded systems hardware needs; typical and advanced, timing diagrams, memories (RAM, ROM, EPROM). Tristate devices, Buses, DMA, UART and PLD's. Built-ins on the microprocessor.	**8**
II.	Interrupts basics, ISR;Context saving, shared data problem. Atomic and critical section, Interrupt latency. Survey of software architectures, Round Robin, Function queue scheduling architecture, Use of real time operating system.	**8**
III.	RTOS, Tasks, Scheduler, Shared data reentrancy, priority inversion, mutex binary semaphore and counting semaphore. Inter task communication, message queue, mailboxes and pipes, timer functions, events. Interrupt routines in an RTOS environment.	**8**
IV.	Embedded system software design using an RTOS. Hard real-time and soft real time system principles, Task division, need of interrupt routines, shared data.	**8**
V.	Embedded Software development tools. Host and target systems, cross compilers, linkers, locators for embedded systems. Getting embedded software in to the target system. Debugging techniques. Testing on host machine, Instruction set emulators, logic analysers. In-circuit emulators and monitors. Regional	**8**

Text Books/Reference:

- John Davies, MSP430 Microcontroller Basics, Elsevier, 2008.
- Andrew N. Sloss et.al. ARM System Developers Guide, ELSEVIER
- Muhammad Ali Mazidi et.al., The 8051 Microcontroller & Embedded Systems, Pearson
- Embedded System Design, A Unified Hardware/Software Introduction, Frank Vahid / Tony Givargis, 2006 reprint, John Wiley Student Edition.

CSE605-P Embedded System Design Lab.

Course Objectives

1

Upon successful completion of the course, students will be able to design simple embedded systems and develop related software. Students also learn to work in a team environment and communicate the results as written reports and oral presentations.

Suggested Microcontroller Platform: Texas Instruments MSP430, ARM 9,
68HC12, 8051.

It is assumed that there are 14 weeks in the semester and about 5 to 6 experiments will be carried out. More experiments are provided to bring in variation.

Experiment #0

Get familiar with the microcontroller kit and the development software. Try the sample programs that are supplied to get familiar with the Microcontroller.

Experiment #1

a) Blink an LED which is connected to your microcontroller using the built-in timer in the icrocontroller. Assume that the LED should be on for x milliseconds and off for y milliseconds; assume that these values are stored in memory locations X and Y. We should be able to change the value of x and y
and rerun the program.

b) Consider an alternate way to program this application. Here, the microcontroller turns the LED on and waits in a busy loop to implement a delay of x milliseconds. Then it turns the LED off and waits in a busy loop to implement a delay of y milliseconds. How do you compare these two solutions?

Experiment #2

Assume that in Experiment #1, the values of x and y have been chosen to be 200 and 500 respectively. When the LED blinking program runs, pressing a key on the keyboard should generate an interrupt to the microcontroller. If the key that has been pressed is a numeric key, the value of x and y must be interchanged by the interrupt service routine. If the key that has been pressed is not a numeric key, then the LED must be turned off for 2 seconds before resuming the blinking.

Experiment #3

If your microcontroller kit has an LCD interface, write a program to display a character string on the LCD. Assume that the string is stored at a location STRING and consists of alphanumeric characters. The string is nullterminated. Modify your program to scroll the displayed string from left to right.

Experiment #4

Modern microcontrollers usually have an in-built Digital-to-Analog and Analogto- Digital converter. Use the built-in DAC to generate voltage waveforms such as (a) pulse train (b) triangular waveform (c) sinusoidal waveform. Observe these waveforms on an oscilloscope.

Experiment #5

Your microcontroller may have a built-in temperature sensor. If not, interface an external temperature sensor to the microcontroller. Write a program to take several measurements of temperature at regular intervals and display the average temperature on the LCD display. Test if the readings change when the ambient temperature changes.

Experiment #6

Your microcontroller may have a built-in ADC. Build a voltmeter that can measure stable voltages in a certain range. The measured value must be displayed on the LCD display. Measure the same voltage using a multimeter and record the error in measurement. Tabulate the error for several values of

	the voltage. **Experiment #7** Build a simple security device based on the microcontroller kit. Interface an external motion sensor to the microcontroller. An alarm must be generated if motion is sensed in a specified region. There must be a provision to record the time at which the intrusion was detected. Similarly, there must be a provision to turn the alarm off by pressing a key. **Experiment #8** A voltage waveform v(t) is available as an input to the microcontroller. We must continuously check the waveform and record the maximum value of the waveform and display the maximum value on the LCD display. Test the program by using a DC supply to generate v(t) and varying the DC value.

CSE606-IADVANCE TOPICS IN OPERATING SYSTEMS

UNIT	CONTENTS	CONTACT HOURS
I.	Operating system structures – policies & mechanism, Structures- monolithic, layered, virtual machines, micro kernel, exokernels, client- server model. Examples from Linux & Windows. Threads Advance Concepts– Libraries- Pthreads, win32 threads, Java threads, Introduction to threading issues, system calls, cancellation, signal handling thread pool, thread specific data, window threads, Linux threads, Solaris Threads. Massage Passing System – Need of Message Passing Systems, design issues, naming, synchronization, Implementation buffering and delivery; mailboxes RPC & RMI. Examples Systems – Linux, Windows	**8**
II.	File System- file system layouts, file system implementation contagious allocation, link list allocation, indexed allocation, file allocation table, virtual file system, directory implementation- linear list and hash table. File System reliability and integrity. I/O system: device drivers/ controllers, busses and interfaces- USB, IDE, SCSI, IEEE1394, RAID system, disk caching and buffering, disk management-disk formatting, RAID Structure, boot block, bad block, swap-space management. System Security: Security Problems, Program Threats, System Network Threats, Cryptography as a Security Tool, User Authentication Implementing security Defenses, Firewalling to Protect Systems and Network, Computer Security Classifications. Overview of security in Windows.	**8**
III.	The Linux OS: Unix Vs Linux, Design Principles, Kernel Structure, components Kernel Modules, Shell- usage, types; An overview of- Process Management, Thread Management and Scheduling, Memory Management, Process Scheduling in Linux File System structure & implementation, I/O Management, Network File System Inter-process Communications Booting and login process, security.	**8**
IV.	The Window OS: Design Principles, System Components- Hardware Abstraction layer, Kernel, Executives; Environmental Subsystems- MS-DOS Environment, 16-bit Windows Environment, Win32 API, POSIX subsystem; IV Exception and Interrupts; An overview of-memory management, process management and thread; Process Scheduling in Windows; File Systems: Internal Layout, recovery, Volume Management and Fault Tolerance, FAT andNTFS, Security features, window registry, OS organizations.	**8**
V.	Multiprocessor Operating Systems: Architecture of Multiprocessor Systems, Overview of Multiprocessor OS, Kernal Structure and Multiprocessing support in Linux & Windows, Process Synchronization- Queued Lock, Spin Lock, Sleep Lock; Process Scheduling. Multimedia Operating System- Introduction to Multimedia & Data Compression- concepts, common graphics file formats, common audio file formats; Video server, Process management- real time scheduling; Multimedia file systems, Multimedia file storage mechanisms, Video sever organization. Mobile Operating System- Windows CE, Palm OS, Symbian OS, JAVA card, Multos.	**8**

Text Books/Reference:

- DM Dhamdhere: Operating Systems – A Concepts Based Approach, Tata McGraw Hill
- Achyut S Godbole: Operating Systems, Tata McGraw Hill
- Tanenbaum: Modern Operating System, Prentice Hall
- A. Silberschatz and Peter B Galvin: Operating System Principals, Wiley India Pvt. Ltd.
- Charles Crowly: Operating System A Design – Oriented Approach, Tata McGraw Hill.
- Bach, Design of Unix Operating Systems.

CSE606-IIARTIFICIAL INTELLIGENCE		
UNIT	**CONTENTS**	**CONTACT HOURS**
I.	Meaning and definition of artificial intelligence, Various types of production systems, Characteristics of production systems Study and comparison of breadth first search and depth first search. Techniques other Search Techniques like hill Climbing, Best first Search. A* algorithm, AO* algorithms etc, and various types of control strategies.	**8**
II.	Knowledge Representation, Problems in representing knowledge, knowledge representionusing propositional and predicate logic, comparison of propositional and predicate logic, Resolution, refutation, deduction, theorem proving, inferencing, monotonic and nonmonotonic reasoning.	**8**
III.	Probabilistic reasoning, Baye's theorem, semantic networks scripts schemas, frames, conceptual dependency and fuzzy logic, forward and backward reasoning.	**8**
IV.	Game playing techniques like minimax procedure, alpha-beta cut-offsetc. planning, Study of the block world problem in robotics Introduction to understanding and natural languages processing.	**8**
V.	Introduction to learning, Various techniques used in learning, introduction to neural networks, applications of neural networks, common sense, reasoning some example of expert systems.	**8**
Text Books/Reference:		
• Artificial Intelligence: Elaine Rich, Kevin Knight, Mc-Graw Hill. • Introduction to AI & Expert System: Dan W. Patterson, PHI. • Artificial Intelligence by Luger (Pearson Education) • Russel & Norvig, Artificial Intelligence: A Modern Approach, Prentice-Hall		

CSE606-IIIHUMAN COMPUTER INTERFACE		
UNIT	**CONTENTS**	**CONTACT HOURS**
I.	The Human: input-output channels, Human memory, thinking, emotions,individual differences, psychology and the design of interactive systems.The Computer: Text entry devices with focus on the design of key boards,positioning, pointing and drawing, display devices.The Interaction: Models of interaction, ergonomics, interaction styles, elementsof WIMP interfaces, interactivity, experience, engagement and fun.Paradigms for Interaction.	**8**
II.	Design Process: The process of design, user focus, scenarios, navigation designscreen design and layout, iteration & prototyping.Usability EngineeringDesign rules: Principles to support usability, standards, guidelines, rules andheuristics, HCI patterns.	**8**
III.	Evaluation Techniques: Definition and goals of evaluation, evaluation throughexpert analysis and user participation, choosing an evaluation method.User support, requirement, approaches, adaptive help systems, designing usersupport systems	**8**
IV.	Cognitive methods: Goals and task hierarchies, linguistic models, challenges ofdisplay based systems, physical and device models, cognitive architectures.	**8**
V.	Communications and collaborations models: Face to Face communication,conversations, Text based communication, group working.Task Analysis: Differences between task analysis and other techniques, task decomposition, knowledge based analysis, ER based analysis, sources ofinformation and data collection, use of task analysis.	**8**
Text Books/Reference:		
• Human Computer Interaction; Alan Dix et.al, 3rd ed., Pearson		

CSE607-P JAVA PROGRAMMING LAB

Objectives: At the end of the semester, the students should have clearly understoodand implemented the following

1	Develop an in depth understanding of programming in Java: data types,variables, operators, operator precedence, Decision and control statements, arrays, switch statement, Iteration Statements, Jump Statements, Using break, Using continue, return.
2	Write Object Oriented programs in Java: Objects, Classes constructors, returningand passing objects as parameter, Inheritance, Access Control, Using super, final with inheritance Overloading and overriding methods, Abstract classes, Extended classes.
3	Develop understanding to developing packages & Interfaces in Java: Package,concept of CLASSPATH, access modifiers, importing package, Defining and implementing interfaces.
4	Develop understanding to developing Strings and exception handling: Stringconstructors, special string operations, character extraction, searching and comparing strings, string Buffer class. Exception handling fundamentals, Exception types, uncaught exceptions, try, catch and multiple catch statements. Usage of throw, throws and finally.
5	Develop applications involving file handling: I/O streams, File I/O.
6	Develop applications involving concurrency: Processes and Threads, ThreadObjects, Defining and Starting a Thread, Pausing Execution with Sleep, Interrupts, Joins, and Synchronization.
7	Develop applications involving Applet: Applet Fundamentals, using paint methodand drawing polygons.

It is expected that each laboratory assignments to given to the students with an aim to In order to achieve the above objectives

Indicative List of exercises:

8	Programs to demonstrate basic concepts e.g. operators, classes, constructors, control & iteration statements, recursion etc. such as complex arithmetic, matrix arithmetic, tower of Hanoi problem etc
9	Development of programs/projects to demonstrate concepts like inheritance, exception handling, packages, interfaces etc. such as application for electricity department, library management, ticket reservation system, payroll system etc
10	Development of a project to demonstrate various file handling concepts.
11	Development of a project to demonstrate various applet concepts

CSE608-P Humanities and Social Sciences

UNIT	CONTENTS	CONTACT HOURS
I.	India-brief history of Indian constitution ,framing-features fundamental rights,duties,directive principles of states,History of Indian National movement,Socio economic growth after independence.	**8**
II.	Society-Social groups-concepts and types,socialization-concept theory,social control:concept,social problem in contempory India,status and role.	**8**
III.	The fundamental of Economics-meaning,definition animportance of economics,Logic of choice,central economic problems,positive and normative approaches,economic systemssocialism and capitalism.	**8**
IV.	Microeconomics-Law of demand and supply,utility approach,indifferencecurves,elasticity of demand & supply and applications,consumer surplus,Law of returns to factors and returns to scale.	**8**
V.	Macroeconomics- concept relating to National product-National income and its measurement,simple Keynesian theory,simple multiplier, money and banking.Meaning,concept of international trade,determination of exchange rate,Balance of payments.	**8**

Text Books/Reference:

- Economics-Lipsey & Chrystal, Oxford Univ.Press,2010Nordhaus, William, Samuelson,Paul-2009-10

CSE701 Cloud Computing

UNIT	CONTENTS	CONTACT HOURS
I.	Introduction Cloud Computing: Nutshell of cloud computing, Enabling Technology, Historicaldevelopment, Vision, feature Characteristics and components of Cloud Computing. Challenges, Risks andApproaches of Migration into Cloud. Ethical Issue in Cloud Computing, Evaluating the Cloud's BusinessImpact and economics, Future of the cloud. Networking Support for Cloud Computing. Ubiquitous Cloudand the Internet of Things	**8**
II.	Cloud Computing Architecture: Cloud Reference Model, Layer and Types of Clouds, Services models,Data center Design and interconnection Network, Architectural design of Compute and Storage Clouds.Cloud Programming and Software: Fractures of cloud programming, Parallel and distributed programmingparadigms-MapReduce, Hadoop , High level Language for Cloud. Programming of Google App engine,	**8**
III.	Virtualization Technology: Definition, Understanding and Benefits of Virtualization. ImplementationLevel of Virtualization, Virtualization Structure/Tools and Mechanisms , Hypervisor VMware, KVM, Xen.Virtualization: of CPU, Memory, I/O Devices, Virtual Cluster and Resources Management, Virtualizationof Server , Desktop, Network, and Virtualization of data-center	**8**
IV.	Securing the Cloud : Cloud Information security fundamentals, Cloud security services, Design principles,Policy Implementation, Cloud Computing Security Challenges, Cloud Computing Security Architecture .Legal issues in cloud Computing.Data Security in Cloud: Business Continuity and Disaster Recovery , Risk Mitigation , Understanding andIdentification of Threats in Cloud, SLA-Service Level Agreements, Trust Management	**8**
V.	*Cloud Platforms in Industry:* Amazon web services , Google AppEngine, Microsoft Azure Design, Aneka:Cloud Application Platform -Integration of Private and Public Clouds*Cloud applications:* Protein structure prediction, Data Analysis, Satellite Image Processing, CRM and ERP,Social networking . Cloud Application- Scientific Application, Business Application.*Advance Topic in Cloud Computing:* Federated Cloud/InterCloud, Third Party Cloud Services	**8**

Text Books/Reference:

- " Distributed and Cloud Computing " By Kai Hawang , Geofrey C.Fox, Jack J. Dongarra Pub: Elservier
- Cloud Computing ,Principal and Paradigms, Edited By Rajkumar Buyya, Jemes Broberg, A. Goscinski, Pub.- Wiley
- Kumar Saurabh, "Cloud Computing" , Wiley Pub
- Krutz , Vines, "Cloud Security " , Wiley Pub
- Velte, "Cloud Computing- A Practical Approach" ,TMH Pub

CSE702 Information System Security

UNIT	CONTENTS	CONTACT HOURS
I.	Introduction to security attacks, services and mechanism, classicalencryption techniques- substitution ciphers and transpositionciphers, cryptanalysis, stream and block ciphers.Modern Block Ciphers: Block ciphers principals, Shannon's theoryof confusion and diffusion, fiestal structure, data encryptionstandard(DES), differential and linear cryptanalysis of DES, blockcipher modes of operations, triple DES.	**8**
II.	AES, RC6, random number generation.S-box theory: Boolean Function, S-box design criteria, Bentfunctions, Propagation and nonlinearity, construction of balancedfunctions, S-box design.	**8**
III.	Public Key Cryptosystems: Principles of Public Key Cryptosystems,RSA Algorithm, security analysis of RSA, Exponentiation inModular Arithmetic.Key Management in Public Key Cryptosystems: Distribution of Public Keys, Distribution of Secret keys using Public KeyCryptosystems. X.509Discrete Logarithms, Diffie-Hellman Key Exchange.	**8**
IV.	Message Authentication and Hash Function: Authenticationrequirements, authentication functions, message authentication code,hash functions, birthday attacks, security of hash functions andMAC, MD5 message digest algorithm, Secure hashalgorithm(SHA).Digital Signatures: Digital Signatures, authentication protocols, digital signature standards (DSS), proof of digital signaturealgorithm.Remote user Authentication using symmetric and AsymmetricAuthentication	**8**
V.	Pretty Good Privacy. IP Security: Overview, IP Security Architecture, AuthenticationHeader, Encapsulation Security Payload in Transport and Tunnelmode with multiple security associations (Key Management notIncluded).Strong Password Protocols: Lamport's Hash, Encrypted KeyExchange.	**8**

Text Books/Reference:

- Stalling Williams: Cryptography and Network Security: Principles and Practices, 4th Edition, Pearson Education, 2006.
- Kaufman Charlie et.al; Network Security: Private Communication in a Public World, 2nd Ed., PHI/Pearson.
- Pieprzyk Josef and et.al; Fundamentals of Computer Security, Springer-Verlag, 2008.
- Trappe & Washington, Introduction to Cryptography, 2nd Ed. Pearson.

CSE703 Data Mining & Ware Housing

UNIT	CONTENTS	CONTACT HOURS
I.	Overview, Motivation(for Data Mining),Data Mining-Definition & Functionalities, Data Processing, Form of Data Preprocessing, Data Cleaning: Missing Values, Noisy Data, (Binning, Clustering, Regression, Computer and Human inspection), Inconsistent Data, Data Integration and Transformation. Data Reduction:-Data Cube Aggregation, Dimensionality reduction, Data Compression, Numerosity Reduction, Clustering, Discretization and Concept hierarchy generation.	**8**
II.	Concept Description: Definition, Data Generalization, Analytical Characterization, Analysis of attribute relevance, Mining Class comparisons, Statistical measures in large Databases. Measuring Central Tendency, Measuring Dispersion of Data, Graph Displays of Basic Statistical class Description, Mining Association Rules in Large Databases, Association rule mining, mining Single-Dimensional Boolean Association rules from Transactional Databases– Apriori Algorithm, Mining Multilevel Association rules from Transaction Databases and Mining Multi- Dimensional Association rules from Relational Databases.	**8**
III.	What is Classification & Prediction, Issues regarding Classification and prediction, Decision tree, Bayesian Classification, Classification by Back propagation, Multilayer feed-forward Neural Network, Back propagation Algorithm, Classification methods K-nearest neighbour classifiers, Genetic Algorithm. Cluster Analysis: Data types in cluster analysis, Categories of clustering methods, Partitioning methods. Hierarchical Clustering-CURE and Chameleon. Density Based Methods-DBSCAN, OPTICS. Grid Based Methods- STING, CLIQUE. Model Based Method –Statistical Approach, Neural Network approach, Outlier Analysis	**8**
IV.	Data Warehousing: Overview, Definition, Delivery Process, Difference between Database System and Data Warehouse, Multi Dimensional Data Model, Data Cubes, Stars, Snow Flakes, Fact Constellations, Concept hierarchy, Process Architecture, 3 Tier Architecture, Data Mining.	**8**
V.	Aggregation, Historical information, Query Facility, OLAP function and Tools. OLAP Servers, ROLAP, MOLAP, HOLAP, Data Mining interface, Security, Backup and Recovery, Tuning Data Warehouse, Testing Data Warehouse.	**8**

Text Books/Reference:

- Data Warehousing in the Real World – Anahory and Murray, Pearson Education.
- Data Mining – Concepts and Techniques – Jiawai Han and Micheline Kamber.
- Building the Data Warehouse – WH Inmon, Wiley.

CSE704 COMPUTER AIDED DESIGN FOR VLSI

UNIT	CONTENTS	CONTACT HOURS
I.	Complexity in microelectronic circuit design and Moore's Law, design styles –Fullcustomdesign, standard-cell design, Programmable Logic Devices, FieldProgrammable Gate Arrays, Design Stages, Computer-Aided Synthesis andOptimizations, design flow and related problems.	**8**
II.	Boolean functions and its representations – co-factor, unite, derivatives,consensus and smoothing; tabular representations and Binary DecisionDiagram (BDD), OBDD, ROBDD and Bryant's reduction algorithm and ITEalgorithm. Hardware abstract models – structures and logic networks, Statediagram, data-flow and sequencing graphs, hierarchical sequencing graphs.Compilation and behavioral optimizations.	**8**
III.	Architectural Synthesis – Circuit description and problem definition, temporal andspatial domain scheduling, synchronization problem. Scheduling algorithms – ASAPand ALAP scheduling algorithms, scheduling under constraints, relative scheduling,list scheduling heuristic. Scheduling in pipelined circuits.	**8**
IV.	Resource Sharing & Binding in sequencing graphs for resource dominated circuits,sharing of registers and busses; binding variables to registers.Two-level logic optimization principles – definitions and exact logic minimizations.Positional cube notations, functions with multi-valued logic. List-orientedmanipulations.	**8**
V.	Physical Design. Floor planning – goals and objectives. Channel definition, I/O andpower planning. Clock Planning. Placement – goals and objectives. Placementalgorithms. Iterative improvement algorithms. Simulated Annealing. Timing-drivenPlacement. Global routing – goals and objectives. Global routing methods. Timingdrivenglobal routing. Detailed Routing – goals and objectives. Left-edge algorithm.Constraints and routing graphs. Channel routing algorithms. Via minimization.Clock routing, power routing, circuit extraction and Design Rule Checking.	**8**

Text Books/Reference:

- S.H. Gerez. Algorithms VLSI Design Automation. Wiley India. (Indian edition available.)
- Michael John Sebastian Smith. Application-Specific Integrated Circuits. Addison-Wesley(Low-priced edition is available.)
- G.D. Micheli, Synthesis and optimization of digital circuits, TMH
- http://www.fie-conference.org/fie98/papers/1002.pdf
- S. Sait and H. Youssef. VLSI Physical Design Automation: Theory and Practice

CSE704-P VLSI PHYSICAL DESIGN LAB

1	VLSI Physical Design Automation is essentially the research, development and productization of algorithms and data structures related to the physical design process. The objective is to investigate optimal arrangements of devices on a plane (or in three dimensions) and efficient interconnection schemes between these devices to obtain the desired functionality and performance. Since space on a wafer is very expensive real estate, algorithms must use the space very efficiently to lower costs and improve yield. In addition, the arrangement of devices plays a key role in determining the performance of a chip. Algorithms for physical design must also ensure that the layout generated abides by all the rules required by the fabrication process. Fabrication rules establish the tolerance limits of the fabrication process. Finally, algorithms must be efficient and should be able to handle verylarge designs. Efficient algorithms not only lead to fast turn-around time, but also permit designers to make iterative improvements to the layouts. The VLSI physical design process manipulates very simple geometric objects, such as polygons and lines. As a result, physical design algorithms tend to be very intuitive in nature, and have significant overlap with graph algorithms and combinatorial optimization algorithms. In view of this observation, many consider physical design automation the study of graph theoretic and combinatorialalgorithms for manipulation of geometric objects in two and three dimensions. However, a pure geometric point of view ignores the electrical (both digital and analog) aspect of the physical design problem. In a VLSI circuit, polygons and lines have inter-related electrical properties, which exhibit a very complex behavior and depend on a host of variables. Therefore, it is necessary to keep the electrical aspects of the geometric objects in perspectivewhile developing algorithms for VLSI physical design automation. With the introduction of Very Deep Sub-Micron (VDSM), which provides very small features and allows dramatic increases in the clock frequency, the effect of electrical parameters on physical design will play a more dominant role in the design and development of new algorithms. (Source: Algorithms For VLSI Physical Design Automation, by Naveed A. Sherwani). **The exercise should be such that the above objectives are met.** Automation tools such as Synopsis/ Cadence are available in the area. However, to begin, the students shall be assigned exercises on route optimization, placement & floor planning. Small circuits may be taken & algorithms implemented. At a later stage, the students may use tools and design more complex circuits.

CSE705 COMPILER DESIGN		
UNIT	**CONTENTS**	**CONTACT HOURS**
I.	Compiler, Translator, Interpreter definition, Phase of compiler introduction to onepass &Multipass compilers, Bootstrapping, Review of Finite automata lexical analyzer,Input, buffering, Recognition of tokens, Idea about LEX: A lexical analyzergenerator, Error handling.	**8**
II.	Review of CFG Ambiguity of grammars, Introduction to parsing. Bottom upparsing Topdown parsing techniques, Shift reduce parsing, Operator precedence parsing,Recursive descent parsing predictive parsers. LL grammars & passers error handlingof LL parser. LR parsers, Construction of SLR, Conical LR & LALR parsing tables,parsing with ambiguous grammar. Introduction of automatic parser generator:YACC error handling in LR parsers.	**8**
III.	Syntax directed definitions; Construction of syntax trees, L-attributeddefinitions, Topdown translation. Specification of a type checker, Intermediate code forms usingpostfix notation and three address code, Representing TAC using triplesand quadruples, Translation of assignment statement. Boolean e xpression and control structures.	**8**
IV.	Storage organization, Storage allocation, Strategies, Activation records,Accessing localand non local names in a block structured language, Parameters passing,Symbol table organization, Data structures used in symbol tables.	**8**
V.	Definition of basic block control flow graphs, DAG representation of basicblock, Advantages of DAG, Sources of optimization, Loop optimization, Ideaabout global data flow analysis, Loop invariant computation, Peephole optimization,Issues in design of code generator, A simple code generator, Code generation fromDAG.	**8**
Text Books/Reference:		
• Aho, Ullman and Sethi: Compilers, Addison Wesley. • Holub, Compiler Design in C, PHI.		

CSE705-P COMPILER DESIGN LAB	
1	Develop an in depth understanding of system programming concept. Lexical analysis, syntax analysis, semantics analysis, code optimization, code generation. Language specification and processing
2	Develop an Understanding of Scanning by using concept of Finite state automaton. Parse tree and syntax tree, Top down parsing (recursive decent parsing, LL (1) parser) Bottom up parsing (operator precedence parsing) .Managing symbol table, opcode table, literal table, pool table
3	Develop an Understanding of Intermediate code form: Three address code, Polish notation (Postfix strings)
4	Develop an Understanding of Allocation data structure. Heaps
5	Develop an Understanding about Language processor development tools: LEX, YACC. Language processing activities (Program generation and execution)
It is expected that each laboratory assignments to given to the students with an aim to In order to achieve the above objectives **Indicative List of exercises:**	
1	Write grammar for a fictitious language and create a lexical analyzer for the same
2	Develop a lexical analyzer to recognize a few patterns in PASCAL and C (ex: identifiers, constants, comments, operators etc.)
3	Develop a lexical analyzer to recognize a few patterns in PASCAL and C (ex: identifiers, constants, comments, operators etc.)
4	Develop on LL (1) parser (Construct parse table also).
5	Develop an operator precedence parser (Construct parse table also)
6	Develop a recursive descent parser
7	Write a program for generating for various intermediate code forms i) Three address code ii) Polish notation
8	Write a program to simulate Heap storage allocation strategy
9	Generate Lexical analyzer using LEX
10	Generate YACC specification for a few syntactic categories
11	Given any intermediate code form implement code optimization techniques
Text Books/Reference:	
• V.V Das, Compiler Design using FLEX and YACC, PHI.	

CSE706-IADVANCE DATABASE MANGEMENT SYSTEMS		
UNIT	**CONTENTS**	**CONTACT HOURS**
I.	Query Processing and Optimization: Overview of Relational Query Optimization,System Catalog in a Relational DBMS, Alternative Plans, Translating SQL, Queriesinto Algebra, Estimating the Cost of a Plan, Relational Algebra Equivalences,Enumeration of Alternative Plans.	**8**
II.	Object Database Systems: Motivating Examples, Structured Data Types, OperationsOn Structured Data, Encapsulation and ADT's, Inheritance, Objects, OIDs andReference Types, Database Design for an ORDBMS, ORDBMS ImplementationChallenges, ORDBMS, Comparing RDBMS, OODBMS, and ORDBMS	**8**
III.	Parallel and Distributed Databases: Architectures for Parallel, Databases, ParallelQuery Evaluation, Parallelizing Individual Operations, Parallel Query Optimization,Distributed DBMS Architectures, Storing Data in a Distributed DBMS, DistributedCatalog Management, Distributed Query Processing, Updating Distributed Data,Introduction to Distributed Transactions, Distributed Concurrency Control,Distributed Recovery.	**8**
IV.	Database Security and Authorization: Introduction to Database Security, AccessControl, Discretionary Access Control- Grant and Revoke on Views and IntegrityConstraints, Mandatory Access Control- Multilevel Relations and Polyinstantiation,Covert Channels, DoD Security Levels, Additional Issues Related to Security- Role ofthe Database Administrator, Security in Statistical Databases, Encryption.	**8**
V.	POSTGES: POSTGRES user interfaces, sql variations and extensions, TransactionManagement, Storage and Indexing, Query processing and optimizations, SystemArchitectures.XML: Motivation, Structure of XML data, XML Document Schema, Querying andTransformation, Application Program Interface to XML, Storage of XML Data, XMLapplications.	**8**
Text Books/Reference:		
• Elmasri R and Navathe SB, Fundamentals of Database Systems, 3rd Edition, Addison Wesley,2000. • Connolly T, Begg C and Strachan A, Database Systems, 2nd Edition, Addison Wesley, 1999 • Ceri Pelagatti , Distributed Database: Principles and System - (McGraw Hill) • Simon AR, Strategic Database Technology: Management for the Year 2000, Morgan Kaufmann,1995 • A. Silversatz, H. Korth and S. Sudarsan: Database Cocepts 5th edition, Mc-Graw Hills 2005.		

CSE706-IIRobotics		
UNIT	**CONTENTS**	**CONTACT HOURS**
I.	Introduction -- brief history, types, classification and usage, Science and Technology of robots, Someuseful websites, textbooks and research journals.	**8**
II.	Elements of robots -- joints, links, actuators, and sensors Position and orientation of a rigid body,Homogeneous transformations, Representation of joints, link representation using D-H parameters,Examples of D-H parameters and link transforms, different kinds of actuators – stepper, DC servo andbrushless motors, model of a DC servo motor, Types of transmissions, Purpose of sensors, internal andexternal sensors, common sensors – encoders, tachometers, strain gauge based force-torque sensors	**8**
III.	Introduction, Direct and inverse kinematics problems, Examples of kinematics of common serialmanipulators, workspace of a serial robot, Inverse kinematics of constrained and redundant robots, Tractrixbased approach for fixed and free robots and multi-body systems, simulations and experiments, Solutionprocedures using theory of elimination, Inverse kinematics solution for the general 6R serial manipulator.	**8**
IV.	Degrees-of-freedom of parallel mechanisms and manipulators, Active and passive joints, Constraint andloop-closure equations, Direct kinematics problem, Mobility of parallel manipulators, Closed-from andnumerical solution, Inverse kinematics of parallel manipulators and mechanisms, Direct kinematics ofGough-Stewart platform.	**8**
V.	Linear and angular velocity of links, Velocity propagation, Manipulator Jacobians for serial and parallelmanipulators, Velocity ellipse and ellipsoids, Singularity analysis for serial and parallel manipulators, Loss and gain of degree of freedom, Statics of serial and parallel manipulators, Statics and force transformation matrix of a Gough-Stewart platform, Singularity analysis and statics.	**8**
Text Books/Reference:		
• Mittal and Nagrath, Robotics and Control, Tata McGraw-Hill Education, 2003 • Fred G. Martin, Robotic Explorations: A Hands On Introduction to Engineering, Pearson Education, 2001.		

CSE706-IIIData Compression Techniques		
UNIT	**CONTENTS**	**CONTACT HOURS**
I.	Compression Techniques: Lossless, lossy, measure of performance, modeling & coding. Lossless compression: Derivation of average information, data models, uniquely decodable codes with tests, prefix codes, Kraft-Mc Millan inequality. Huffman coding: Algorithms, minimum variance Huffman codes, optimality, length extended codes, adaptive coding, Rice codes, using Huffman codes for lossless image compression.	**8**
II.	Arithmetic coding with application to lossless compression. Dictionary Techniques: LZ77, LZ78, LZW Predictive coding: Burrows-Wheeler Transform and move-to-front coding, JPEG-LS Facsimile Encoding: Run length, T.4 and T.6	**8**
III.	Lossy coding- Mathematical preliminaries: Distortion criteria, conditional entropy, average mutual information, differential entropy, rate distortion theory, probability and linear system models. Scalar quantization: The quantization problem, uniform quantizer, Forward adaptive quantization, non-uniform quantization-Formal adopting quantization, companded Quantization Vector quantization: Introduction, advantages, The Linde-Ruzo-Grey algorithm, lattice vector quantization.	**8**
IV.	Differential encoding – Introduction, Basic algorithm, Adaptive DPCM, Delta modulation, speech and image coding using delta modulation. Sampling in frequency and time domain, z-transform, DCT, DST, DWHT, quantization and coding of transform coefficient.	**8**
V.	Sub band coding: Introduction, Filters, Basic algorithm, Design of Filter banks, G.722, MPEG. Wavelet based compression: Introduction, wavelets multi-resolution analysis and the scaling function implementation using filters.	**8**
Text Books/Reference:		
• Sayood K: Introduction to Data Compression: ELSEVIER 2005.		

CSE707- P Web Development Lab	
1	Creation of HTML Files
2	Working with Client Side Scripting : VBScript, JavaScript
3	Configuration of web servers: Apache Web Server, Internet Information Server (IIS)
4	Working with ActiveX Controls in web documents
5	Experiments in Java Server Pages: Implementing MVC Architecture using Servlets, Data Access Programming (using ADO), Session and Application objects, File System Management
6	Working with other Server Side Scripting: Active Server Pages, Java Servlets, PHP
7	Experiments in Ajax Programming
8	Developing Web Services
9	Developing any E-commerce application (Mini Project)
10	Application Development in cloud computing Environment
11	Experiment Using Open Source Tool e.g. ANEKA

CSE801 MOBILE COMPUTING

UNIT	CONTENTS	CONTACT HOURS
I.	Mobile computing: Definitions, adaptability issues (transparency, Environmental Constraints, application aware adaptation), mechanisms for adaptation and incorporating adaptations. Mobility management: mobility management, location management principle and techniques, PCS location management Scheme.	**8**
II.	Data dissemination and management: challenges, Data dissemination, bandwidth allocation for publishing, broadcast disk scheduling, mobile cache maintenance schemes, Mobile Web Caching. Introduction to mobile middleware.	**8**
III.	Middleware for application development: adaptation, Mobile agents. Service Discovery Middleware: Service Discovery & standardization Methods (universally Unique Identifiers, Textual Description & using interfaces), unicast Discovery, Multicast Discovery & advertisement, service catalogs, Garbage Collection, Eventing.	**8**
IV.	Mobile IP, Mobile TCP, Database systems in mobile environments, World Wide Web and mobility	**8**
V.	Ad Hoc networks, localization, MAC issues, Routing protocols, global state routing (GSR), Destination sequenced distance vector routing (DSDV), Dynamic source routing (DSR), Ad Hoc on demand distance vector routing (AODV), Temporary ordered routing algorithm (TORA), QoS in Ad Hoc Networks, applications.	**8**

Text Books/Reference:

- Frank Adelstein, Sandeep Gupta, Golden Richard III, Loren Schwiebert, Fundamentals of Mobile andPervasive Computing, TMH.
- Principles of mobile computing Hansmann & Merk., Springer
- Mobile communications Jochen Schiller , Pearson
- 802.11 wireless networks Matthew S.Gast, O'REILLY
- Wireless LANs: Davis & McGuffin, McGraw Hill
- Mobile Communications Handbook by Jerry D. Gybson
- Mobile Communications Handbook by R

CSE802 Digital Image Processing		
UNIT	**CONTENTS**	**CONTACT HOURS**
I.	**Introduction to Image Processing:** Digital Image representation, Sampling &Quantization, Steps in image Processing, Image acquisition, color imagerepresentation	**8**
II.	**Image Transformation & Filtering:** Intensity transform functions, histogramprocessing, Spatial filtering, Fourier transforms and its properties, frequency domainfilters, colour models, Pseudo colouring, colour transforms, Basics of WaveletTransforms	**8**
III.	**Image Restoration:** Image degradation and restoration process, Noise Models, NoiseFilters, degradation function, Inverse Filtering, Homomorphism Filtering	**8**
IV.	**Image Compression:** Coding redundancy, Interpixel redundancy, Psychovisualredundancy, Huffman Coding, Arithmetic coding, Lossy compression techniques,JPEG Compression	**8**
V.	**Image Segmentation & Representation:** Point, Line and Edge Detection,Thresholding, Edge and Boundary linking, Hough transforms, Region BasedSegmentation, Boundary representation, Boundary Descriptors, Regional	**8**

Text Books/Reference:

- Gonzalez and Woods: Digital Image Processing ISDN 0-201-600- 781, Addison Wesley 1992. Boyle and Thomas: Computer Vision - A First Gurse 2nd Edition, ISBN 0-632-028-67X, Blackwell Science 1995.
- Gonzalez and Woods: Digital Image Processing ISDN 0-201-600- 781, Addison Wesley 1992
- Pakhera Malay K: Digital Image Processing and Pattern Recogination, PHI.
- Trucco&Verri: Introductory Techniques for 3-D Computer Vision, Prentice Hall, Latest Edition
- Low: Introductory Computer Vision and Image Processing, McGraw-Hill 1991, ISBN 0-07-707403-3

CSE802- P Digital Image Processing lab	
1	Color image segmentation algorithm development
2	Wavelet/vector quantization compression
3	Deformable templates applied to skin tumor border finding
4	Helicopter image enhancement
5	High-speed film image enhancement
6	Computer vision for skin tumor image evaluation
7	New Border Images

.

CSE803 DISTRIBUTED SYSTEMS

UNIT	CONTENTS	CONTACT HOURS
I.	Distributed Systems: Features of distributed systems, nodes of a distributed system, Distributedcomputation paradigms, Model of distributed systems, Types of Operating systems:Centralized Operating System, Network Operating Systems, Distributed Operating Systemsand Cooperative Autonomous Systems, design issues in distributed operating systems.Systems Concepts and Architectures: Goals, Transparency, Services, Architecture Models,Distributed Computing Environment (DCE).[1.2]Theoretical issues in distributed systems: Notions of time and state, states and events in adistributed system, time, clocks and event precedence, recording the state of distributedsystems.	**8**
II.	Concurrent Processes and Programming: Processes and Threads, Graph Models for ProcessRepresentation, Client/Server Model, Time Services, Language Mechanisms forSynchronization, Object Model Resource Servers, Characteristics of Concurrent ProgrammingLanguages (Language not included).[1]Inter-process Communication and Coordination: Message Passing, Request/Reply andTransaction Communication, Name and Directory services, RPC and RMI case studies.[1]	**8**
III.	Distributed Process Scheduling: A System Performance Model, Static Process Scheduling withCommunication, Dynamic Load Sharing and Balancing, Distributed ProcessImplementation.[1]Distributed File Systems: Transparencies and Characteristics of DFS, DFS Design andimplementation, Transaction Service and Concurrency Control, Data and FileReplication.[1,2]Case studies: Sun network file systems, General Parallel file System and Window's filesystems. Andrew and Coda File Systems [2,3]	**8**
IV.	Distributed Shared Memory: Non-Uniform Memory Access Architectures, MemoryConsistency Models, Multiprocessor Cache Systems, Distributed Shared Memory,Implementation of DSM systems.[1]Models of Distributed Computation: Preliminaries, Causality, Distributed Snapshots, Modelinga Distributed Computation, Failures in a Distributed System, Distributed Mutual Exclusion,Election, Distributed Deadlock handling, Distributed termination detection. [1]	**8**
V.	Distributed Agreement: Concept of Faults, failure and recovery, Byzantine Faults, Adversaries,Byzantine Agreement, Impossibility of Consensus and Randomized Distributed Agreement.[1]Replicated Data Management: concepts and issues, Database Techniques, Atomic Multicast,and Update Propagation.[1] CORBA case study: Introduction, Architecture, CORBA RMI, CORBA Services.[3]	**8**

Text Books/Reference:

- Distributed operating systems and algorithm analysis by Randy Chow and T. Johnson, Pearson
- Operating Systems A concept based approach by DM Dhamdhere, TMH
- Distributed Systems- concepts and Design, Coulouris G., Dollimore J, and Kindberg T., Pearson

CSE804-IHardware Testing and Fault Tolerance

UNIT	CONTENTS	CONTACT HOURS
I.	Overview of hardware testing. Reliability and Testing, Difference between Verification andTesting, Concepts of fault models, test pattern generation and fault coverage. Types of tests– exhaustive testing, pseudo-exhaustive testing, pseudo-random testing, and deterministictesting. Test Application. Design for Test. Testing Economics. Defects, Failures and Faults.How are physical defects modeled as faults. Stuck-at faults, Single stuck-at-faults multiplestuck-at faults, bridging faults, delay faults, transient faults.	**8**
II.	Relation between VLSI Design and Testing. a) Design Representation for the purpose of testing – Representation in the form ofmathematical equations, tabular format, graphs, Binary Decision Diagrams, Netlists, orHDL descriptions. b) Recap of VLSI Design Flow and where testing fits in the flow. Importance ofSimulation and Fault Simulation. Compiled and event-driven simulation. Parallel anddeductive fault simulation. Using fault simulation to estimate fault coverage andbuilding a fault dictionary	**8**
III.	Combinational Test Pattern Generation. D-algorithm. Critical Path Tracking. PODEMalgorithm for test generation. Testing sequential circuits. Functional and deterministic ATPGfor sequential circuits and the associated challenges. Motivation for Design for Testability.Test Points, Partitioning for Testability. Scan Testing. Scan Architectures. Cost of ScanTesting. Boundary Scan Testing. Board-level testing. Boundary-scan Architecture andvarious modes of operation.	**8**
IV.	a) Built-in Self Test. Pseudo-random test generation. Response Compaction. Randompattern-resistant faults. BIST architectures – Circular BIST, BILBO, STUMPS. b) Testing of Memories – Fault models, Functional tests for memories, Memory BIST. c) Testing of microprocessors.	**8**
V.	Hardware fault tolerance. Failure Rate, Reliability, Mean Time to Failure. Different kinds ofredundancy schemes for fault-tolerance (Space, Time, and Information Redundancy). NmodularRedundancy. Watch Dog Processors, Byzantine Failures. Information Redundancy– parity codes, checksums, m-of-n codes. RAID architectures for disk storage systems. Faulttolerance in interconnection networks. Fault-tolerant routing techniques.	**8**

Text Books/Reference:

- Samiha Mourad and Yervant Zorian. Principles of Electronic Systems. Wiley Student
- Koren and C. Mani Krishna. Fault-Tolerant Systems. Elsevier. (Indian Edition Available.)
- Abramovici, M., Breuer, M. A. and Friedman, A. D. Digital systems testing and testable design. IEEE press (Indian edition available through Jayco Publishing house), 2001.2.

Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI Circuits by

CSE804-IIREAL TIME SYSTEMS

UNIT	CONTENTS	CONTACT HOURS
I.	Introduction: Definition, Typical Real Time Applications, concept of tasks, types of tasks and real time systems, block diagram of RTS, and tasks parameters –Release Times, execution time, period, Deadlines, and Timing Constraints etc. RTS requirements.	**8**
II.	Reference Models for Real Time Systems: processors and Resources, Temporal Parameters of Real-Time Workload, Periodic and Aperiodic Task Model, Precedence Constrains and Data Dependency, Other Types of Dependencies, Functional Parameters, Resource Parameters. Real Time Scheduling: classification of Real Time Scheduling, scheduling criteria, performance metrics, schedulability analysis, Introduction to Clock Driven scheduling, Weighted Round Robin Approach and Priority Driven Approach. Dynamic Versus Static systems, Offline Versus Online Scheduling.	**8**
III.	Periodic tasks scheduling: Clock Driven Scheduling – definition, notations and assumption, scheduler concepts, general scheduling structure, cyclic executives. Priority Driven Scheduling; notations and assumption, fixed priority verses dynamic priority, fixed priority scheduling algorithms (RM and DM) and their schedulability analysis, concept of schedulability tests – Inexact and exact schedulability tests for RM and DM, Optimality of the RM and DM algorithms, practical factors.	**8**
IV.	Aperiodic task scheduling; assumption and approaches, server based and non-server based fixed priority scheduling algorithms – polling server, deferrable server , simple sporadic server, priority exchange, extended priority exchange, slack stealing. Introduction to scheduling of flexible computations –flexible applications, imprecise computation model and firm deadline model.	**8**
V.	Resources Access Control: Assumptions on Resources and their usage, Effect of Resource Contention and Resource Access Control (RAC), Non-preemptive Critical Sections, priority inversion problem, need of new resource synchronization primitives/protocols for RTS, Basic Priority-Inheritance and Priority-Ceiling Protocols, Stack Based Priority-Ceiling Protocol, Use of Priority- Ceiling Protocol in Dynamic Priority Systems, Preemption Ceiling Protocol, Access Control in Multiple- Unit Resources, Controlling Concurrent Accesses to Data Objects.	**8**

Text Books/Reference:

- J.W.S.Liu: Real-Time Systems, Pearson Education Asia
- P.D.Laurence, K.Mauch: Real-time Microcomputer System Design, An Introduction, McGraw Hill
- C.M. Krisna & K. G. Shim- Real time systems- TMH

CSE804-IIIInformation Retrieval		
UNIT	**CONTENTS**	**CONTACT HOURS**
I.	Knowledge Representation: Knowledge representation, Basics of Prepositional logic,Predicate logic, reasoning using first order logic, unification, forward chaining,backward chaining, resolution Production rules, frames, semantic networks scripts.	**8**
II.	Ontology Development: Description logic-taxonomies, Topic maps Ontology,Definition expressing ontology, logically ontology representations, – XML, RDF,RDFS, OWL, OIL, ontology development for specific domain, ontology engineering,Semantic web services.	**8**
III.	Information Retrieval Modeling: Information retrieval, taxonomy, formalcharacterization, classic information retrieval, set theoretic model, algebraic model,probabilistic model, structured text, retrieval models, models for browsing, retrievalperformance evaluation, keyword based querying, pattern matching, structuralqueries, query operations.	**8**
IV.	Text and Multimedia Languages and Properties: Introduction, metadata, markuplanguages, multimedia. Text operations: document preprocessing, documentclustering text Compressionbasic concepts - statistical methods. Indexing andsearching: inverted files, suffix trees, signature file, Boolean queries, sequentialsearching, pattern matching.	**8**
V.	Recent Trends in IR: Parallel and distributed IR, multimedia IR, data modeling, querylanguages, A generic Multimedia indexing Approach, one dimensional time series,two dimensional color images, Automatic feature extraction. Web Searching,Characterizing the Web, Search Engines, Browsing, Meta searchers, Searching using hyperlinks	**8**

Text Books/Reference:

- Stuart Russell and Peter Norvig, “Artificial Intelligence – A Modern Approach”, Pearson Education, Second edition, 2003. (UNIT I)
- Michael C. Daconta, Leo J. Obart and Kevin J. Smith,”Semantic Web – A Guide to the Future of XMLWeb Services and Knowledge Management”, Wiley Publishers, 2003 (UNIT II)
- Ricardo Baeza-Yates, BerthierRibeiro-Neto, “Modern Information Retrieval”, Addison Wesley, 1999(UNITs III, IV & V)
- Elain Rich and Kevin Knight, "Artificial Intelligence", Tata McGraw-Hill, Third edition, 2003
- Christopher D. Manning, PrabhakarRaghavan and HinrichSchutze, “Introduction to Information Retrieval”, Cambridge University Press, 2008.

CSE805-P UNIX NETWORK PROGRAMMING & SIMULATION LAB

Objectives:

At the end of course, the students should be able to

- Understand various distributions of Unix viz. BSD, POSIX etc.
- Write client/server applications involving unix sockets involving TCP or UDPinvolving iterative or concurrent server.
- Understand IPV4 & IPV6 interoperability issues
- Use fork( ) system call
- Understand the network simulator NS2 and Simulate routing algorithm on NS2 (Available on http://www.isi.edu/nsnam/ns/).

Suggested Platform: For Socket Programming- Linux, For NS2 Any of Microsoft Windowsor Linux (In case of Microsoft, Virtual environment cygwin will also be required).

Suggested Exercises

1	Write two programs in C: hello_client and hello_server • The server listens for, and accepts, a single TCP connection; it reads all the data itcan from that connection, and prints it to the screen; then it closes the connection • The client connects to the server, sends the string “Hello, world!”, then closes theconnection
2	Write an Echo_Client and Echo_server using TCP to estimate the round trip time from client to the server. The server should be such that it can accept multiple connections at any given time.
3	Repeat Exercises 1 & 2 for UDP.
4	Repeat Exercise 2 with multiplexed I/O operations
5	Simulate Bellman-Ford Routing algorithm in NS2

Text Books/Reference:

- Stevens, Unix Network Programming, Vol-I