

1.3.2. Average percentage of courses that include experiential learning through project work/field work/internship during last five years

ACADEMIC YEAR 2017-18

COMPUTER SCIENCE AND ENGINEERING

Document contains list and syllabus of courses that includes experiential learning through project works and internships.

Documents Enclosed

Sl. No	Particulars	Page Nos
1.	List of course that include experiential learning through project works, internships and field work	1-2
2.	Syllabus of the courses mapped	3-111

LIST OF COURSE THAT INCLUDE EXPERIENTIAL LEARNING

Total mapped course : 52+12

SL NO	COURSE CODE	COURSE NAME
1.	10PHY12/10PHY22	ENGINEERING PHYSICS
2.	10CCP13/10CCP23	COMPUTER CONCEPTS AND C PROGRAMMING
3.	10ELE15/ 10ELE25	BASIC ELECTRICAL ENGINEERING
4.	10ELN15 / 10ELN25	BASIC ELECTRONICS
5.	10CPL16 / 10CPL26	COMPUTER PROGRAMMING LABORATORY
6.	10PHYL17/10PHYL27	LABORATORY EXPERIMENTS IN ENGINEERING PHYSICS
7.	10CIP18/10CIP28	CONSTITUTION OF INDIA AND PROFESSIONAL ETHICS
8.	10CS32	ELECTRONIC CIRCUITS
9.	10CS33	LOGIC DESIGN
10.	10CS34	DISCRETE MATHEMATICAL STRUCTURES
11.	10CS35	DATA STRUCTURES WITH C
12.	10CS36	OBJECT ORIENTED PROGRAMMING WITH C++
13.	10CSL37	DATA STRUCTURES WITH C/C++ LABORATORY
14.	10CSL38	ELECTRONIC CIRCUITS & LOGIC DESIGN LABORATORY
15.	10CS42	GRAPH THEORY AND COMBINATORICS
16.	10CS43	DESIGN AND ANALYSIS OF ALGORITHMS
17.	10CS44	UNIX AND SHELL PROGRAMMING
18.	10CS45	MICROPROCESSORS
19.	10CS46	COMPUTER ORGANIZATION
20.	10CSL47	DESIGN AND ANALYSIS OF ALGORITHMS LABORATORY
21.	10CSL48	MICROPROCESSORS LABORATORY
22.	10IS51	SOFTWARE ENGINEERING
23.	10CS52	SYSTEMS SOFTWARE
24.	10CS53	OPERATING SYSTEMS
25.	10CS54	DATABASE MANAGEMENT SYSTEMS
26.	10CS55	COMPUTER NETWORKS - I
27.	10CSL57	DATABASE APPLICATIONS LABORATORY
28.	10CSL58	SYSTEMS SOFTWARE & OPERATING SYSTEMS LABORATORY
29.	10AL61	MANAGEMENT AND ENTREPRENEURSHIP
30.	10CS62	UNIX SYSTEM PROGRAMMING
31.	10CS64	COMPUTER NETWORKS - II
32.	10CS65	COMPUTER GRAPHICS AND VISUALIZATION
33.	10CS661/10IS661	OPERATIONS RESEARCH
34.	10CSL67	COMPUTER GRAPHICS AND VISUALIZATION LABORATORY
35.	10CSL68	UNIX SYSTEM PROGRAMMING AND COMPILER DESIGN LABORATORY

36.	10CS663	DATA COMPRESSION
37.	10CS71	OBJECT-ORIENTED MODELING AND
38.	10CS72/ 10IS752	EMBEDDED COMPUTING SYSTEMS
39.	10CS73	PROGRAMMING THE WEB
40.	10CS753/10IS753	JAVA AND J2EE
41.	10CS761/10IS761	C# PROGRAMMING AND .NET
42.	10CS755	DATA WAREHOUSING AND DATA MINING
43.	10CSL77	NETWORKS LABORATORY
44.	10CSL78	WEB PROGRAMMING LABOR
45.	10CS762	DIGITAL IMAGE PROCESSING
46.	10CS82	SYSTEM MODELING AND SIMULATION
47.	10CS835/10IS835	INFORMATION AND NETWORK SECURITY
48.	10CS842	SOFTWARE TESTING
49.	10CS845/10IS845	CLOUDS, GRIDS AND CLUSTERS
50.	10CS85	PROJECT WORK
51.	10CS832	WEB 2.0 AND RICH INTERNET APPLICATIONS
52.	10CS765/10IS765	STORAGE AREA NETWORKS
53.	16SCS12	CLOUD COMPUTING
54.	16SCS13	ADVANCES IN DATA BASE MANAGEMENT SYSTEM
55.	16SCS21	MANAGING BIG DATA
56.	16SCS22	ADVANCES IN COMPUTER NETWORKS
57.	16SCS24	INTERNET OF THINGS
58.	16SCS253	INFORMATION AND NETWORK SECURITY
59.	16SCS31	SEMINAR / PRESENTATION ON INTERNSHIP
60.	16SCS32	REPORT ON INTERNSHIP
61.	16SCS34	EVALUATION OF PROJECT PHASE -1
62.	16SCS41	MACHINE LEARNING TECHNIQUES
63.	16SCS43	EVALUATION OF PROJECT PHASE -2
64.	16SCS44	EVALUATION OF PROJECT AND VIVA-VOCE

Syllabus of the courses mapped:

ENGINEERING PHYSICS

Sub Code	: 10PHY12/10PHY22	IA Marks	: 25
Hrs/ Week	: 04	Exam Hours	: 03
Total Hrs.	: 52	Exam Marks	: 100

PART – A

UNIT-1

Modern Physics

Introduction to Blackbody radiation spectrum, Photo-electric effect, Compton effect. Wave particle Dualism. de Broglie hypothesis – de Broglie wavelength, extension to electron particle. – Davisson and Germer Experiment.

Matter waves and their Characteristic properties. Phase velocity, group velocity and Particle velocity. Relation between phase velocity and group velocity. Relation between group velocity and particle velocity. Expression for deBroglie wavelength using group velocity.

7 Hours

UNIT-2

Quantum Mechanics

Heisenberg's uncertainty principle and its physical significance. Application of uncertainty principle (Non-existence of electron in the nucleus, Explanation for β -decay and kinetic energy of electron in an atom). Wave function. Properties and Physical significance of a wave function. Probability density and Normalisation of wave function. Setting up of a one dimensional, time independent Schrödinger wave equation. Eigen values and Eigen functions. Application of Schrödinger wave equation – Energy Eigen values for a free particle. Energy Eigen values of a particle in a potential well of infinite depth.

6 Hours

UNIT-3

Electrical Conductivity in Metals

Free-electron concept. Classical free-electron theory - Assumptions. Drift velocity. Mean collision time and mean free path. Relaxation time. Expression for drift velocity. Expression for electrical conductivity in metals. Effect of impurity and temperature on electrical resistivity of metals. Failures of classical free-electron theory.

Quantum free-electron theory - Assumptions. Fermi - Dirac Statistics. Fermi-energy – Fermi factor. Density of states (No derivation). Expression for electrical resistivity / conductivity. Temperature dependence of resistivity of metals. Merits of Quantum free – electron theory.

7 Hours

UNIT-4

Dielectric & Magnetic Properties of Materials

Dielectric constant and polarisation of dielectric materials. Types of polarisation. Equation for internal field in liquids and solids (one dimensional). Clausius – Mossotti equation. Ferro and Piezo – electricity (qualitative). Frequency dependence of dielectric constant. Important applications of dielectric materials. Classification of dia, para and ferromagnetic materials. Hysteresis in ferromagnetic materials. Soft and Hard magnetic materials. Applications.

7 Hours

PART – B

UNIT - 5

Lasers

Principle and production. Einstein's coefficients (expression for energy density). Requisites of a Laser system. Condition for Laser action. Principle, Construction and working of He-Ne and semiconductor Laser. Applications of Laser – Laser welding, cutting and drilling. Measurement of atmospheric pollutants. Holography – Principle of Recording and reconstruction of 3-D images. Selected applications of holography.

6 Hours

UNIT-6

Optical Fibers & Superconductivity

Propagation mechanism in optical fibers. Angle of acceptance. Numerical aperture. Types of optical fibers and modes of propagation. Attenuation. Applications – block diagram discussion of point to point communication.

Temperature dependence of resistivity in superconducting materials. Effect of magnetic field (Meissner effect). Type I and Type II superconductors - Temperature dependence of critical field. BCS theory (qualitative). High temperature superconductors. Applications of superconductors– Superconducting magnets, Maglev vehicles and squids

7 Hours

UNIT-7

Crystal Structure

Space lattice, Bravais lattice - unit cell, primitive cell. Lattice parameters. Crystal systems. Direction and planes in a crystal. Miller indices. Expression for inter-planar spacing. Co-ordination number. Atomic packing factor. Bragg's Law. Determination of crystal structure by Bragg's x-ray spectrometer. Crystal structures of NaCl, and diamond.

6 Hours

UNIT-8**Material Science**

Introduction to Nanoscience and Nanotechnology. Nanomaterials: Shapes of nanomaterials, Methods of preparation of nanomaterials, Wonders of nanotechnology: Discovery of Fullerene and carbon nanotubes, Applications. Ultrasonic non-destructive testing of materials. Measurements of velocity in solids and liquids, Elastic constants.

6 Hours**Text Books**

	Title		Author/s / Editor		Publishers
1	Solid State Physics – Sixth Edition	-	S.O. Pillai	-	New Age International
2	Engineering Physics	-	V. Rajendran	-	Tata Mc-Graw Hill Company Ltd., New Delhi

Reference Books

	Title		Author/s / Editor		Publishers
1	Nanosystems- Molecular Machinery, Manufacturing and Computation	-	K.Eric Drexler	-	John Wiley & Sons 2005 Ed.
2	Fundamentals and Applications of Ultrasonic Waves	-	J David N Cheeke and Cheeke N Cheeke	-	CRC Press
3	Nano Materials	-	Vishwanathan	-	Narosa Publications
4	Engineering Physics	-	G.K Shivakumar	-	Prism Books Pvt. Ltd.

COMPUTER CONCEPTS AND C PROGRAMMING

Subject Code	: 10CCP13/10CCP23	IA Marks	: 25
Hrs/Week	: 04	Exam Hours	: 03
Total Hrs.	: 52	Exam Marks	: 100

PART – A

UNIT-1

Introduction to Computer Systems, Interacting with the Computer, Computer Organization

The Computer defined, Early history, Basic parts and structure of a computer, Categorizing Computers, Information Processing life cycle, Essential computer hardware, Essential computer software.

Keyboard, Mouse, Inputting data in other ways: Pen-based systems, Data scanning devices, Game controllers, Voice recognition devices, Microphone, Visual input devices, Video and sound, Monitors, Printers, Plotters, Data projectors, Sound systems. Number systems, ASCII, BCD, CPU, Buses, Mother Board, Chip sets, Microprocessors.

7 Hours

UNIT-2

Storage Device Concepts, Operating Systems, Networking

Storage media, Floppy drive, Hard disks, Optical media, CD-ROM, CD-R, CD-RW, DVD-ROM, Recordable DVD.

Software, Custom-made Software, Shrunken-wrapped software, Types of operating systems, Computer processing techniques, Functions of Operating Systems, Management of processor, Memory, Virtual storage, devices, and information.

Networking, Convergence of computing with communications, Networking basics, Need for networking, Basic components of a network.

7 Hours

UNIT-3

Fundamentals of Problem Solving, Introduction to C Language

Creating and running programs, System development, Software Engineering. Introduction to C Language: Background, C Programs, Identifiers, Types, Variables, Constants, Input / Output, Programming example, Software Engineering, Tips and common programming errors.

6 Hours

UNIT-4

Structure of a C Program

Expressions, Precedence and associativity, Side effects, Evaluating expressions, Type conversion, Statements, Programming examples, Software Engineering, Tips and common programming errors.

6 Hours

PART - B

UNIT-5

Functions

Designing structured programs, Functions in C, User-defined Functions, Inter-function communication, Standard functions, Scope, Programming examples, Software Engineering, Tips and common programming errors.

6 Hours

UNIT-6

Selection – Making Decisions, Repetition

Logical data and operators, Two-way selection, Multiway-selection, Concept of a loop, pre-test and post-test loops, Initialization and updating, Event controlled and count controlled loops, Loops in C, Other statements related to looping, looping applications, Recursion, Programming examples, Software Engineering, Tips and common programming errors.

7 Hours

UNIT-7

Arrays, Strings

Concepts, Using arrays in C, Inter-function communication, Array applications, Bubble Sort, Binary search, Two-dimensional Arrays, Multi-dimensional arrays, String concepts, C strings, String input/output, Programming examples, Software Engineering, Tips and common programming errors.

7 Hours

UNIT-8

Basic Concepts of Parallel Programming

Motivating parallelism, Scope for parallel computing, Thread basics, Why threads? OpenMP: A standard for directive – based parallel programming, The OpenMP programming model, Specifying concurrent tasks in OpenMP, Synchronization constructs in OpenMP, Data handling in OpenMP, OpenMP library functions, Environment variables in OpenMP.

6 Hours

Text Books:

1. Vikas Gupta: Computer Concepts and C Programming, Dreamtech Press/Wiley India, 2009.
2. Behrouz A. Forouzan, Richard F. Gilberg: Computer Science - A Structured Approach Using C, 3rd Edition, Cengage Learning, 2007.
3. Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar: Introduction to Parallel Computing, 2nd Edition, Pearson Education, 2003.

Reference Books:

1. Peter Norton: Introduction to Computers, 7th Edition, Tata McGraw Hill, 2010.
2. E. Balagurusamy: Programming in ANSI C, 4th Edition, Tata McGraw Hill, 2008.
3. Brian W. Kernighan and Dennis Ritchie: The C Programming Language, 2nd Edition, PHI, 1998.

Web Reference: <http://elearning.vtu.ac.in>

BASIC ELECTRICAL ENGINEERING

Sub Code	:	10ELE15/ 10ELE25	IA Marks	:	25
Hrs/ Week	:	04	Exam Hours	:	03
Total Hrs.	:	52	Exam Marks	:	100

PART – A

UNIT-1

1-a) D. C. Circuits: Ohm's Law and Kirchoff's Laws, analysis of series, parallel and series- parallel circuits excited by independent voltage sources. Power and Energy. Illustrative examples.

4Hours

1-b) Electromagnetism: Faradays Laws, Lenz's Law, Fleming's Rules, Statically and dynamically induced emf's. Concept of self inductance, mutual inductance and coefficient of coupling. Energy stored in magnetic field. Illustrative examples.

3Hours

UNIT-2

2.Single-phase A.C. Circuits: Generation of sinusoidal voltage, definition of average value, root mean square value, form factor and peak factor of sinusoidally varying voltage and current, phasor representation of alternating quantities. Analysis, with phasor diagrams, of R, L, C, R-L, R-C and R-L-C circuits, real power, reactive power, apparent power and power factor. Illustrative examples involving series, parallel and series- parallel circuits.

7 Hours

UNIT-3

3 Three Phase Circuits: Necessity and advantages of three phase systems, generation of three phase power, definition of Phase sequence, balanced supply and balanced load. Relationship between line and phase values of balanced star and delta connections. Power in balanced three-phase circuits, measurement of power by two-wattmeter method. Illustrative examples.

6 Hours

UNIT-4

4-a) Measuring Instruments: Construction and Principle of operation of dynamometer type wattmeter and single-phase induction type energy meter (problems excluded).

3 Hours

4-b) Domestic Wiring: Service mains, meter board and distribution board. Brief discussion on Cleat, Casing & Capping and conduit (concealed) wiring. Two-way and three-way control of a lamp. Elementary discussion on fuse and Miniature Circuit Breaker (MCB's). Electric shock, precautions against shock –Earthing: Pipe and Plate.

3 Hours

PART – B

UNIT-5

5.DC Machines: Working principle of DC machine as a generator and a motor. Types and constructional features. emf equation of generator, relation between emf induced and terminal voltage enumerating the brush drop and drop due to armature reaction. Illustrative examples. DC motor working principle, Back emf and its significance, torque equation. Types of D.C. motors, characteristics and applications. Necessity of a starter for DC motor. Illustrative examples on back emf and torque.

7 Hours

UNIT-6

6. Transformers: Principle of operation and construction of single-phase transformers (core and shell types). emf equation, losses, efficiency and voltage regulation (Open Circuit and Short circuit tests, equivalent circuit and phasor diagrams are excluded). Illustrative problems on emf equation and efficiency only.

7 Hours

UNIT-7

7. Synchronous Generators: Principle of operation. Types and constructional features. emf equation. Concept of winding factor (excluding derivation of distribution and pitch factors). Illustrative examples on emf equation.

6 Hours

UNIT-8

8. Three Phase Induction Motors: Concept of rotating magnetic field. Principle of operation. Types and Constructional features. Slip and its significance. Applications of squirrel - cage and slip - ring motors. Necessity of a starter, star-delta starter. Illustrative examples on slip calculations.

6 Hours

Text Books:

1. “Basic Electrical Engineering”, D C Kulshreshtha, ,TMH,2009 Edition.
2. “Fundamentals of Electrical Engineering”, Rajendra Prasad, PHI, Second Edition, 2009.

Reference Books:

- 1 "Electrical Technology", E. Hughes International Students 9th Edition, Pearson, 2005.
- 2 "Basic Electrical Engineering", Abhijit Chakrabarti, Sudipta Nath, Chandan Kumar Chanda, TMH, First reprint 2009.
- 3 Problems in Electrical Engineering, Parker Smith, CBS Publishers and Distributors, 9th Edition, 2003.

BASIC ELECTRONICS

Sub Code	: 10ELN15 / 10ELN25	IA Marks	: 25
Hrs/ Week	: 04	Exam Hours	: 03
Total Hrs.	: 52	Exam Marks	: 100

PART - A

UNIT-1

SEMICONDUCTOR DIODES AND APPLICATIONS: p-n junction diode, Characteristics and Parameters, Diode approximations, DC load line, Temperature dependence of p-n characteristics, AC equivalent circuits, Zener diodes Half-wave diode rectifier, Ripple factor, Full-wave diode rectifier, Other full-wave circuits, Shunt capacitor filter - Approximate analysis of capacitor filters, Power supply performance, Zener diode voltage regulators, Numerical examples as applicable.

7 Hours

UNIT-2

TRANSISTORS: Bipolar Junction transistor, Transistor Voltages and currents, amplification, Common Base, Common Emitter and Common Collector Characteristics, DC Load line and Bias Point.

6 Hours

UNIT-3

BIASING METHODS: Base Bias, Collector to Base Bias, Voltage divider Bias, Comparison of basic bias circuits, Bias circuit design, Thermal Stability of Bias Circuits (Qualitative discussions only).

6 Hours

UNIT-4

OTHER DEVICES: Silicon Controlled Rectifier (S.C.R), SCR Control Circuits, More S.C.R applications; Unijunction transistor, UJT applications, Junction Field effect Transistors(Exclude Fabrication and Packaging), JFET Characteristics, FET Amplifications, Numerical examples as applicable.

7 Hours

PART - B

UNIT-5

AMPLIFIERS & OSCILLATORS: Decibels and Half power points, Single Stage CE Amplifier and Capacitor coupled two stage CE amplifier(Qualitative discussions only), Series voltage negative feedback and Additional effects of Negative feed back(Qualitative discussions only), The Barkhausen Criterion for Oscillations, BJT RC phase shift oscillator, Hartley, Colpitts and crystal oscillator (Qualitative discussions only) Numerical problems as applicable.

6 Hours

UNIT-6

INTRODUCTION TO OPERATIONAL AMPLIFIERS: Ideal OPAMP, Saturable property of an OP AMP, Inverting and Non Inverting OPAMP circuits, Need for OPAMP, Characteristics and applications - voltage follower, addition, subtraction, integration, differentiation; Numerical examples as applicable Cathode Ray Oscilloscope (CRO).

6 Hours

UNIT-7

COMMUNICATION SYSTEMS: Block diagram, Modulation, Radio Systems, Superhetrodyne Receivers, Numerical examples as applicable

NUMBER SYSTEMS: Introduction, decimal system, Binary, Octal and Hexadecimal number systems, addition and subtraction, fractional number, Binary Coded Decimal numbers.

7 Hours

UNIT-8

DIGITAL LOGIC:, Boolean algebra, Logic gates, Half-adder, Full-adder, Parallel Binary adder.

7 Hours

Text Books:

1. Electronic Devices and Circuits: David. A. Bell; Oxford University Press, 5th Edn., 2008.

Reference Books:

1. Electronic Devices and Circuits: Jacob Millman, Christos C. Halkias TMH, 1991 Reprint 2001.
2. Electronic Communication Systems, George Kennedy, TMH 4th Edition.
3. Digital Logic and Computer Design, Morris Mano, PHI, EEE.
4. Basic Eelctronics, RD Sudhaker Samuel, U B Mahadevaswamy, V. Nattarsu, Saguine-Pearson, 2007.

COMPUTER PROGRAMMING LABORATORY

Subject Code	: 10CPL16 / 10CPL26	IA Marks	: 25
Hrs/Week	: 03	Exam Hours	: 03
Total Hrs.	: 42	Exam Marks	: 50

PART – A

1. Design, develop and execute a program in C to find and output all the roots of a given quadratic equation, for non-zero coefficients.
2. Design, develop and execute a program in C to implement Euclid's algorithm to find the GCD and LCM of two integers and to output the results along with the given integers.
3. Design, develop and execute a program in C to reverse a given four digit integer number and check whether it is a palindrome or not. Output the given number with suitable message.
4. Design, develop and execute a program in C to evaluate the given polynomial $f(x) = a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0$ for given value of x and the coefficients using Horner's method.
5. Design, develop and execute a program in C to copy its input to its output, replacing each string of one or more blanks by a single blank.
6. Design, develop and execute a program in C to input N integer numbers in ascending order into a single dimensional array and perform a binary search for a given key integer number and report success or failure in the form of a suitable message.
7. Design, develop and execute a program in C to input N integer numbers into a single dimensional array, sort them in ascending order using bubble sort technique and print both the given array and the sorted array with suitable headings.
8. Design, develop and execute a program in C to compute and print the word length on the host machine.

PART – B

9. Design, develop and execute a program in C to calculate the approximate value of $\exp(0.5)$ using the Taylor Series expansion for the exponential function. Use the terms in the expansion until the last term is less than the machine epsilon defined FLT_EPSILON in the header file <float.h>. Also print the value returned by the Mathematical function $\exp()$.
10. Design, develop and execute a program in C to read two matrices A (M x N) and B (P x Q) and compute the product of A and B if the matrices are compatible for multiplication. The program must print the input matrices and

the resultant matrix with suitable headings and format if the matrices are compatible for multiplication, otherwise the program must print a suitable message. (For the purpose of demonstration, the array sizes M, N, P, and Q can all be less than or equal to 3)

11. Design, develop and execute a parallel program in C to add, element-wise, two one-dimensional arrays A and B of N integer elements and store the result in another one-dimensional array C of N integer elements.

12. Design and develop a function `rightrot(x, n)` in C that returns the value of the integer x rotated to the right by n bit positions as an unsigned integer. Invoke the function from the main with different values for x and n and print the results with suitable headings.

13. Design and develop a function `isprime(x)` that accepts an integer argument and returns 1 if the argument is prime and 0 otherwise. The function must use plain division checking approach to determine if a given number is prime. Invoke this function from the main with different values obtained from the user and print appropriate messages.

14. Design, develop and execute a parallel program in C to determine and print the prime numbers which are less than 100 making use of algorithm of the Sieve of Eratosthenes.

15. Design and develop a function `reverses(s)` in C to reverse the string s in place. Invoke this function from the main for different strings and print the original and reversed strings.

16. Design and develop a function `match any(s1,s2)` which returns the first location in the string s1 where any character from the string s2 occurs, or - 1 if s1 contains no character from s2. Do not use the standard library function which does a similar job! Invoke the function `match any(s1, s2)` from the main for different strings and print both the strings and the return value from the function `match any(s1,s2)`.

Note: In the practical examination, the student has to answer two questions. One question from Part A and one question from Part B will be selected by the student by lots. All the questions listed in the syllabus have to be included in the lots. The change of question (Part A only / Part B only / Both Part A & Part B) has to be considered, provided the request is made for the same, within half an hour from the start of the examination. The allotment of marks is as detailed below:

Sl. No.	Activity		Max. Marks
1.	Procedure Writing program & procedure for the assigned problems along with algorithms / flowchart	Part A	5*
		Part B	5*
2.	Conduction Execution of the program and	Part A	10

	showing the results in proper format	Part B	20
3.	Viva-voce**		10
Total Max. Marks			50
Minimum passing Marks (40% of Max. Marks)			20

* To be considered as zero if student has been allowed change of question.

LABORATORY EXPERIMENTS IN ENGINEERING PHYSICS

Sub Code	: 10PHYL17/10PHYL27	IA Marks	: 25
Hrs/ Week	: 03	Exam Hours	: 03
Total Hrs.	: 10 (To be completed)	Exam Marks	: 50

EXPERIMENTS :

1. Series & Parallel LCR Circuits.(Determination of resonant frequency & quality factor)
2. I-V Characteristics of Zener Diode.(determination of knee voltage, zener voltage & forward resistance)
3. Characteristics of a Transistor.(Study of Input & Output characteristics and calculation of input resistance, output resistance & amplification factor)
4. Photo Diode Characteristics.(Study of I-V characteristics in reverse bias and variation of photocurrent as a function of reverse voltage & intensity)
5. Ultrasonic Interferometer (Measurement of velocity of sounds in solids/liquids).
6. Dielectric constant (Measurement of dielectric constant).
7. Magnetic properties (Study of retentivity and coercivity by B-H graph method).
8. Diffraction (Measurement of wavelength of laser / Hg source using diffraction grating).
9. Planck's constant (Using the principle of photoelectric effect/LED's).
10. Electrical Resistivity (Determination of resistivity in semiconductor by Four probe method).
11. Verification of Stefan's law.
12. Determination of Fermi energy.(Measurement of Fermi energy in copper)
13. Uniform Bending Experiment.(Determination of Youngs modulus of material bar)
14. Newtons Rings.(Determination of radius of curvature of planoconvex lens)

CONSTITUTION OF INDIA AND PROFESSIONAL ETHICS

Sub Code	: 10CIP18/10CIP28	IA Marks	:	25
Hrs/ Week	: 02	Exam Hours	:	02
Total Hrs.	: 26	Exam Marks	:	50

- 1 Preamble to the constitution of India. Fundamental rights under Part – III – details of Exercise of rights, Limitations & Important cases. **4 Hours**
- 2 Relevance of Directive principles of State Policy under Part – IV. Fundamental duties & their significance. **3 Hours**
- 3 Union Executive – President, Prime Minister, Parliament & the Supreme Court of India. **3 Hours**
- 4 State executive – Governors, Chief Minister, State Legislator and High Courts. **3 Hours**
- 5 Constitutional Provisions for Scheduled Castes & Tribes, Women & Children & Backward classes. Emergency Provisions. **4 Hours**
- 6 Electoral process, Amendment procedure, 42nd, 44th, 74th, 76th, 86th and 91st Constitutional amendments. **3 Hours**
- 7 Scope & aims of engineering Ethics. Responsibility of Engineers. Impediments to responsibility. **3 Hours**
- 8 Honesty, Integrity and reliability, risks, safety & liability in engineering. **3 Hours**

Text Books:

1. Durga Das Basu: “Introduction to the Constitution of India” (Students Edn.) Prentice – Hall EEE, 19th/20th Edn., 2001.
2. “Engineering Ethics” by Charles E.Haries, Michael. S.Pritchard and Michael J.Robins Thompson Asia, 2003-08-05.

Reference Books:

1. “An Introduction to Constitution of India” by M.V.Pylee, Vikas Publishing, 2002.
2. “Engineering Ethics” by M. Govindarajan, S.Natarajan, V.S. Senthilkumar., Prentice – Hall of India Pvt. Ltd. New Delhi, 2004.

Scheme of examination:

Question paper is of objective type. Students have to pass this subject compulsorily. However, marks will not be considered for awarding class/rank.

ELECTRONIC CIRCUITS
(Common to CSE & ISE)

Subject Code: 10CS32
Hours/Week : 04
Total Hours : 52

I.A. Marks : 25
Exam Hours: 03
Exam Marks: 100

PART - A

UNIT - 1 **7 Hours**

Transistors, UJTs, and Thyristors: Operating Point, Common-Emitter Configuration, Thermal Runaway, Transistor Switch, Unijunction Transistors, SCR.

UNIT - 2 **6 Hours**

Field Effect Transistors: Bipolar Junction Transistors versus Field Effect Transistors, Junction Field Effect Transistors, Metal Oxide Field Effect Transistors, Differences between JFETs and MOSFETs, Handling MOSFETs, Biasing MOSFETs, FET Applications, CMOS Devices, Insulated Gate Bipolar Transistors (IGBTs)

UNIT - 3 **6 Hours**

Optoelectronic Devices: Introduction, Photosensors, Photoconductors, Photodiodes, Phototransistors, Light-Emitting Diodes, Liquid Crystal Displays, Cathode Ray Tube Displays, Emerging Display Technologies, Optocouplers

UNIT - 4 **7 Hours**

Small Signal Analysis of Amplifiers: Amplifier Bandwidth: General Frequency Considerations, Hybrid h-Parameter Model for an Amplifier, Transistor Hybrid Model, Analysis of a Transistor Amplifier using complete h-Parameter Model, Analysis of a Transistor Amplifier Configurations using Simplified h-Parameter Model (CE configuration only), Small-Signal Analysis of FET Amplifiers, Cascading Amplifiers, Darlington Amplifier, Low-Frequency Response of Amplifiers (BJT amplifiers only).

PART - B

UNIT - 5 **6 Hours**

Large Signal Amplifiers, Feedback Amplifier: Classification and characteristics of Large Signal Amplifiers, Feedback Amplifiers: Classification of Amplifiers, Amplifier with Negative Feedback, Advantages of Negative Feedback, Feedback Topologies, Voltage-Series (Series-Shunt)

Feedback, Voltage-Shunt (Shunt-Shunt) Feedback, Current-Series (Series-Series) Feedback, Current-Shunt (Shunt-Series) Feedback.

UNIT - 6

7 Hours

Sinusoidal Oscillators, Wave-Shaping Circuits: Classification of Oscillators, Conditions for Oscillations: Barkhausen Criterion, Types of Oscillators, Crystal Oscillator, Voltage-Controlled Oscillators, Frequency Stability.

Wave-Shaping Circuits: Basic RC Low-Pass Circuit, RC Low-Pass Circuit as Integrator, Basic RC High-Pass Circuit, RC High-Pass Circuit as Differentiator, Multivibrators, Integrated Circuit (IC) Multivibrators.

UNIT - 7

7 Hours

Linear Power Supplies, Switched mode Power Supplies: Linear Power Supplies: Constituents of a Linear Power Supply, Designing Mains Transformer; Linear IC Voltage Regulators, Regulated Power Supply Parameters.

Switched Mode Power Supplies: Switched Mode Power Supplies, Switching Regulators, Connecting Power Converters in Series, Connecting Power Converters in Parallel

UNIT - 8

6 Hours

Operational Amplifiers: Ideal Opamp versus Practical Opamp, Performance Parameters, Some Applications: Peak Detector Circuit, Absolute Value Circuit, Comparator, Active Filters, Phase Shifters, Instrumentation Amplifier, Non-Linear Amplifier, Relaxation Oscillator, Current-To-Voltage Converter, Voltage-To-Current Converter, Sine Wave Oscillators.

Text Book:

1. Anil K Maini, Varsha Agarwal: Electronic Devices and Circuits, Wiley, 2009.
(4.1, 4.2, 4.7, 4.8, 5.1 to 5.3, 5.5, 5.6, 5.8, 5.9, 5.13, 5.14, 6.1, 6.3, 7.1 to 7.5, 7.10 to 7.14, Listed topics only from 8, 10.1, 11, 12.1, 12.2, 12.3, 12.5, 13.1 to 13.6, 13.9, 13.10, 14.1, 14.2, 14.6, 14.7, 15.1, 15.5 to 15.7. 16.3, 16.4, 17.12 to 17.22)

Reference Books:

1. Jacob Millman, Christos Halkias, Chetan D Parikh: Millman's Integrated Electronics – Analog and Digital Circuits and Systems, 2nd Edition, Tata McGraw Hill, 2010.
2. R. D. Sudhaker Samuel: Electronic Circuits, Sanguine-Pearson, 2010.

LOGIC DESIGN
(Common to CSE & ISE)

Subject Code: 10CS33
Hours/Week : 04
Total Hours : 52

I.A. Marks : 25
Exam Hours: 03
Exam Marks: 100

PART-A

UNIT – 1 **7 Hours**

Digital Principles, Digital Logic: Definitions for Digital Signals, Digital Waveforms, Digital Logic, 7400 TTL Series, TTL Parameters The Basic Gates: NOT, OR, AND, Universal Logic Gates: NOR, NAND, Positive and Negative Logic, Introduction to HDL.

UNIT – 2 **6 Hours**

Combinational Logic Circuits

Sum-of-Products Method, Truth Table to Karnaugh Map, Pairs Quads, and Octets, Karnaugh Simplifications, Don't-care Conditions, Product-of-sums Method, Product-of-sums simplifications, Simplification by Quine-McClusky Method, Hazards and Hazard Covers, HDL Implementation Models.

UNIT – 3 **6 Hours**

Data-Processing Circuits: Multiplexers, Demultiplexers, 1-of-16 Decoder, Encoders, Exclusive-or Gates, Parity Generators and Checkers, Magnitude Comparator, Programmable Array Logic, Programmable Logic Arrays, HDL Implementation of Data Processing Circuits

UNIT – 4 **7 Hours**

Clocks, Flip-Flops: Clock Waveforms, TTL Clock, Schmitt Trigger, Clocked D FLIP-FLOP, Edge-triggered D FLIP-FLOP, Edge-triggered JK FLIP-FLOP, FLIP-FLOP Timing, JK Master-slave FLIP-FLOP, Switch Contact Bounce Circuits, Various Representation of FLIP-FLOPs, Analysis of Sequential Circuits, HDL Implementation of FLIP-FLOP

PART-B

UNIT – 5 **6 Hours**

Registers: Types of Registers, Serial In - Serial Out, Serial In - Parallel out, Parallel In - Serial Out, Parallel In - Parallel Out, Universal Shift Register, Applications of Shift Registers, Register Implementation in HDL

UNIT – 6 **7 Hours**

Counters: Asynchronous Counters, Decoding Gates, Synchronous Counters, Changing the Counter Modulus, Decade Counters, Presetable Counters,

Counter Design as a Synthesis problem, A Digital Clock, Counter Design using HDL

UNIT – 7

7 Hours

Design of Synchronous and Asynchronous Sequential Circuits: Design of Synchronous Sequential Circuit: Model Selection, State Transition Diagram, State Synthesis Table, Design Equations and Circuit Diagram, Implementation using Read Only Memory, Algorithmic State Machine, State Reduction Technique.

Asynchronous Sequential Circuit: Analysis of Asynchronous Sequential Circuit, Problems with Asynchronous Sequential Circuits, Design of Asynchronous Sequential Circuit, FSM Implementation in HDL

UNIT – 8

6 Hours

D/A Conversion and A/D Conversion: Variable, Resistor Networks, Binary Ladders, D/A Converters, D/A Accuracy and Resolution, A/D Converter-Simultaneous Conversion, A/D Converter-Counter Method, Continuous A/D Conversion, A/D Techniques, Dual-slope A/D Conversion, A/D Accuracy and Resolution

Text Book:

1. Donald P Leach, Albert Paul Malvino & Goutam Saha: Digital Principles and Applications, 7th Edition, Tata McGraw Hill, 2010.

Reference Books:

1. Stephen Brown, Zvonko Vranesic: Fundamentals of Digital Logic Design with VHDL, 2nd Edition, Tata McGraw Hill, 2005.
2. R D Sudhaker Samuel: Illustrative Approach to Logic Design, Sanguine-Pearson, 2010.
3. Charles H. Roth: Fundamentals of Logic Design, Jr., 5th Edition, Cengage Learning, 2004.
4. Ronald J. Tocci, Neal S. Widmer, Gregory L. Moss: Digital Systems Principles and Applications, 10th Edition, Pearson Education, 2007.
5. M Morris Mano: Digital Logic and Computer Design, 10th Edition, Pearson Education, 2008.

DISCRETE MATHEMATICAL STRUCTURES
(Common to CSE & ISE)

Subject Code: 10CS34

Hours/Week : 04

Total Hours : 52

I.A. Marks : 25

Exam Hours: 03

Exam Marks: 100

PART – A

UNIT – 1 **6 Hours**

Set Theory: Sets and Subsets, Set Operations and the Laws of Set Theory, Counting and Venn Diagrams, A First Word on Probability, Countable and Uncountable Sets

UNIT – 2 **7 Hours**

Fundamentals of Logic: Basic Connectives and Truth Tables, Logic Equivalence – The Laws of Logic, Logical Implication – Rules of Inference

UNIT – 3 **6 Hours**

Fundamentals of Logic contd.: The Use of Quantifiers, Quantifiers, Definitions and the Proofs of Theorems

UNIT – 4 **7 Hours**

Properties of the Integers: Mathematical Induction, The Well Ordering Principle – Mathematical Induction, Recursive Definitions

PART – B

UNIT – 5 **7 Hours**

Relations and Functions: Cartesian Products and Relations, Functions – Plain and One-to-One, Onto Functions – Stirling Numbers of the Second Kind, Special Functions, The Pigeon-hole Principle, Function Composition and Inverse Functions

UNIT – 6 **7 Hours**

Relations contd.: Properties of Relations, Computer Recognition – Zero-One Matrices and Directed Graphs, Partial Orders – Hasse Diagrams, Equivalence Relations and Partitions

UNIT – 7 **6 Hours**

Groups: Definitions, Examples, and Elementary Properties, Homomorphisms, Isomorphisms, and Cyclic Groups, Cosets, and Lagrange's Theorem.

Coding Theory and Rings: Elements of Coding Theory, The Hamming Metric, The Parity Check, and Generator Matrices

UNIT – 8

6 Hours

Group Codes: Decoding with Coset Leaders, Hamming Matrices

Rings and Modular Arithmetic: The Ring Structure – Definition and Examples, Ring Properties and Substructures, The Integers Modulo n

Text Book:

1. Ralph P. Grimaldi: Discrete and Combinatorial Mathematics, , 5th Edition, Pearson Education, 2004.
(Chapter 3.1, 3.2, 3.3, 3.4, Appendix 3, Chapter 2, Chapter 4.1, 4.2, Chapter 5.1 to 5.6, Chapter 7.1 to 7.4, Chapter 16.1, 16.2, 16.3, 16.5 to 16.9, and Chapter 14.1, 14.2, 14.3).

Reference Books:

1. Kenneth H. Rosen: Discrete Mathematics and its Applications, 7th Edition, McGraw Hill, 2010.
2. Jayant Ganguly: A Treatise on Discrete Mathematical Structures, Sanguine-Pearson, 2010.
3. D.S. Malik and M.K. Sen: Discrete Mathematical Structures: Theory and Applications, Cengage Learning, 2004.
4. Thomas Koshy: Discrete Mathematics with Applications, Elsevier, 2005, Reprint 2008.

DATA STRUCTURES WITH C (Common to CSE & ISE)

Subject Code: 10CS35

Hours/Week : 04

Total Hours : 52

I.A. Marks : 25

Exam Hours: 03

Exam Marks: 100

PART – A

UNIT - 1

8 Hours

BASIC CONCEPTS: Pointers and Dynamic Memory Allocation, Algorithm Specification, Data Abstraction, Performance Analysis, Performance Measurement

UNIT - 2

6 Hours

ARRAYS and STRUCTURES: Arrays, Dynamically Allocated Arrays, Structures and Unions, Polynomials, Sparse Matrices, Representation of Multidimensional Arrays

UNIT - 3**6 Hours**

STACKS AND QUEUES: Stacks, Stacks Using Dynamic Arrays, Queues, Circular Queues Using Dynamic Arrays, Evaluation of Expressions, Multiple Stacks and Queues.

UNIT - 4**6 Hours**

LINKED LISTS: Singly Linked lists and Chains, Representing Chains in C, Linked Stacks and Queues, Polynomials, Additional List operations, Sparse Matrices, Doubly Linked Lists

PART - B**UNIT - 5****6 Hours**

TREES – 1: Introduction, Binary Trees, Binary Tree Traversals, Threaded Binary Trees, Heaps.

UNIT - 6**6 Hours**

TREES – 2, GRAPHS: Binary Search Trees, Selection Trees, Forests, Representation of Disjoint Sets, Counting Binary Trees, The Graph Abstract Data Type.

UNIT - 7**6 Hours**

PRIORITY QUEUES Single- and Double-Ended Priority Queues, Leftist Trees, Binomial Heaps, Fibonacci Heaps, Pairing Heaps.

UNIT - 8**8 Hours**

EFFICIENT BINARY SEARCH TREES: Optimal Binary Search Trees, AVL Trees, Red-Black Trees, Splay Trees.

Text Book:

1. Horowitz, Sahni, Anderson-Freed: Fundamentals of Data Structures in C, 2nd Edition, Universities Press, 2007.
(Chapters 1, 2.1 to 2.6, 3, 4, 5.1 to 5.3, 5.5 to 5.11, 6.1, 9.1 to 9.5, 10)

Reference Books:

1. Yedidyah, Augenstein, Tannenbaum: Data Structures Using C and C++, 2nd Edition, Pearson Education, 2003.
2. Debasis Samanta: Classic Data Structures, 2nd Edition, PHI, 2009.
3. Richard F. Gilberg and Behrouz A. Forouzan: Data Structures A Pseudocode Approach with C, Cengage Learning, 2005.

4. Robert Kruse & Bruce Leung: Data Structures & Program Design in C, Pearson Education, 2007.

**OBJECT ORIENTED PROGRAMMING WITH C++
(Common to CSE & ISE)**

Subject Code: 10CS36
Hours/Week : 04
Total Hours : 52

I.A. Marks : 25
Exam Hours: 03
Exam Marks: 100

PART – A

UNIT 1

6 Hours

Introduction: Overview of C++, Sample C++ program, Different data types, operators, expressions, and statements, arrays and strings, pointers & user-defined types

Function Components, argument passing, inline functions, function overloading, recursive functions

UNIT 2

7 Hours

Classes & Objects – I: Class Specification, Class Objects, Scope resolution operator, Access members, Defining member functions, Data hiding, Constructors, Destructors, Parameterized constructors, Static data members, Functions

UNIT 3

7 Hours

Classes & Objects –II: Friend functions, Passing objects as arguments, Returning objects, Arrays of objects, Dynamic objects, Pointers to objects, Copy constructors, Generic functions and classes, Applications

Operator overloading using friend functions such as +, -, pre-increment, post-increment, [] etc., overloading <<, >>.

UNIT 4

6 Hours

Inheritance – I: Base Class, Inheritance and protected members, Protected base class inheritance, Inheriting multiple base classes

PART – B

UNIT 5

6 Hours

Inheritance – II: Constructors, Destructors and Inheritance, Passing parameters to base class constructors, Granting access, Virtual base classes

UNIT 6 **7 Hours**
Virtual functions, Polymorphism: Virtual function, Calling a Virtual function through a base class reference, Virtual attribute is inherited, Virtual functions are hierarchical, Pure virtual functions, Abstract classes, Using virtual functions, Early and late binding.

UNIT 7 **6 Hours**
I/O System Basics, File I/O: C++ stream classes, Formatted I/O, I/O manipulators, fstream and the File classes, File operations

UNIT 8 **7 Hours**
Exception Handling, STL: Exception handling fundamentals, Exception handling options
STL: An overview, containers, vectors, lists, maps.

Text Books:

1. Herbert Schildt: The Complete Reference C++, 4th Edition, Tata McGraw Hill, 2003.

Reference Books:

1. Stanley B.Lippmann, Josee Lajore: C++ Primer, 4th Edition, Pearson Education, 2005.
2. Paul J Deitel, Harvey M Deitel: C++ for Programmers, Pearson Education, 2009.
3. K R Venugopal, Rajkumar Buyya, T Ravi Shankar: Mastering C++, Tata McGraw Hill, 1999.

DATA STRUCTURES WITH C/C++ LABORATORY
(Common to CSE & ISE)

Subject Code: 10CSL37	I.A. Marks : 25
Hours/Week : 03	Exam Hours: 03
Total Hours : 42	Exam Marks: 50

1. Using circular representation for a polynomial, design, develop, and execute a program in C to accept two polynomials, add them, and then print the resulting polynomial.
2. Design, develop, and execute a program in C to convert a given valid parenthesized infix arithmetic expression to postfix expression and then to print both the expressions. The expression consists of

single character operands and the binary operators + (plus), - (minus), * (multiply) and / (divide).

3. Design, develop, and execute a program in C to evaluate a valid postfix expression using stack. Assume that the postfix expression is read as a single line consisting of non-negative single digit operands and binary arithmetic operators. The arithmetic operators are + (add), - (subtract), * (multiply) and / (divide).
4. Design, develop, and execute a program in C to simulate the working of a queue of integers using an array. Provide the following operations:
 - a. Insert
 - b. Delete
 - c. Display
5. Design, develop, and execute a program in C++ based on the following requirements:

An EMPLOYEE class is to contain the following data members and member functions:

Data members: Employee_Number (an integer), Employee_Name (a string of characters), Basic_Salary (an integer) , All_Allowances (an integer), IT (an integer), Net_Salary (an integer).

Member functions: to read the data of an employee, to calculate Net_Salary and to print the values of all the data members.
(All_Allowances = 123% of Basic; Income Tax (IT) = 30% of the gross salary (= basic_Salary _ All_Allowance); Net_Salary = Basic_Salary + All_Allowances – IT)
6. Design, develop, and execute a program in C++ to create a class called STRING and implement the following operations. Display the results after every operation by overloading the operator <<.
 - i. STRING s1 = "VTU"
 - ii. STRING s2 = "BELGAUM"
 - iii. STIRNG s3 = s1 + s2; (Use copy constructor)
7. Design, develop, and execute a program in C++ to create a class called STACK using an array of integers and to implement the following operations by overloading the operators + and - :
 - i. s1=s1 + element; where s1 is an object of the class STACK and element is an integer to be pushed on to top of the stack.
 - ii. s1=s1- ; where s1 is an object of the class STACK and - operator pops off the top element.

Handle the STACK Empty and STACK Full conditions. Also display the contents of the stack after each operation, by overloading the operator <<.

8. Design, develop, and execute a program in C++ to create a class called LIST (linked list) with member functions to insert an element at the front of the list as well as to delete an element from the front of the list. Demonstrate all the functions after creating a list object.
9. Design, develop, and execute a program in C to read a sparse matrix of integer values and to search the sparse matrix for an element specified by the user. Print the result of the search appropriately. Use the triple <row, column, value> to represent an element in the sparse matrix.
10. Design, develop, and execute a program in C to create a max heap of integers by accepting one element at a time and by inserting it immediately in to the heap. Use the array representation for the heap. Display the array at the end of insertion phase.
11. Design, develop, and execute a program in C to implement a doubly linked list where each node consists of integers. The program should support the following operations:
 - i. Create a doubly linked list by adding each node at the front.
 - ii. Insert a new node to the left of the node whose key value is read as an input.
 - iii. Delete the node of a given data if it is found, otherwise display appropriate message.
 - iv. Display the contents of the list.(Note: Only either (a,b and d) or (a, c and d) may be asked in the examination)
12. Design, develop, and execute a program in C++ to create a class called DATE with methods to accept two valid dates in the form dd/mm/yy and to implement the following operations by overloading the operators + and -. After every operation the results are to be displayed by overloading the operator <<.
 - i. $\text{no_of_days} = d1 - d2$; where $d1$ and $d2$ are DATE objects, $d1 \geq d2$ and no_of_days is an integer.
 - ii. $d2 = d1 + \text{no_of_days}$; where $d1$ is a DATE object and no_of_days is an integer.
13. Design, develop, and execute a program in C++ to create a class called OCTAL, which has the characteristics of an octal number.

Implement the following operations by writing an appropriate constructor and an overloaded operator +.

- i. `OCTAL h = x` ; where x is an integer
- ii. `int y = h + k` ; where h is an OCTAL object and k is an integer.

Display the OCTAL result by overloading the operator <<. Also display the values of h and y.

14. Design, develop, and execute a program in C++ to create a class called `BIN_TREE` that represents a Binary Tree, with member functions to perform inorder, preorder and postorder traversals. Create a `BIN_TREE` object and demonstrate the traversals.

Note: In the examination each student picks one question from a lot of *all* the 14 questions.

ELECTRONIC CIRCUITS & LOGIC DESIGN LABORATORY (Common to CSE & ISE)

Subject Code: 10CSL38

Hours/Week : 03

Total Hours : 42

I.A. Marks : 25

Exam Hours: 03

Exam Marks : 50

PART-A

1. a) Design and construct a suitable circuit and demonstrate the working of positive clipper, double-ended clipper and positive clamper using diodes.
b) Demonstrate the working of the above circuits using a simulation package.
2. a) Design and construct a suitable circuit and determine the frequency response, input impedance, output impedance, and bandwidth of a CE amplifier.
b) Design and build the CE amplifier circuit using a simulation package and determine the voltage gain for two different values of supply voltage and for two different values of emitter resistance.
3. a) Design and construct a suitable circuit and determine the drain characteristics and transconductance characteristics of an enhancement-mode MOSFET.
b) Design and build CMOS inverter using a simulation package and verify its truth table.

4. a) Design and construct a Schmitt trigger using Op-Amp for given UTP and LTP values and demonstrate its working.
b) Design and implement a Schmitt trigger using Op-Amp using a simulation package for two sets of UTP and LTP values and demonstrate its working.
5. a) Design and construct a rectangular waveform generator (Op-Amp relaxation oscillator) for given frequency and demonstrate its working.
b) Design and implement a rectangular waveform generator (Op-Amp relaxation oscillator) using a simulation package and demonstrate the change in frequency when all resistor values are doubled.
6. Design and implement an astable multivibrator circuit using 555 timer for a given frequency and duty cycle.

PART – B

7. a) Given a 4-variable logic expression, simplify it using Entered Variable Map and realize the simplified logic expression using 8:1 multiplexer IC.
b) Design and develop the Verilog /VHDL code for an 8:1 multiplexer. Simulate and verify its working.
8. a) Realize a J-K Master / Slave Flip-Flop using NAND gates and verify its truth table.
b) Design and develop the Verilog / VHDL code for D Flip-Flop with positive-edge triggering. Simulate and verify its working.
9. a) Design and implement a mod-n ($n < 8$) synchronous up counter using J-K Flip-Flop ICs and demonstrate its working.
b) Design and develop the Verilog / VHDL code for mod-8 up counter. Simulate and verify its working.
10. a) Design and implement a ring counter using 4-bit shift register and demonstrate its working.
b) Design and develop the Verilog / VHDL code for switched tail counter. Simulate and verify its working.
11. Design and implement an asynchronous counter using decade counter IC to count up from 0 to n ($n \leq 9$) and demonstrate its working.

Reference Books:

1. B.V. Ramana: Higher Engineering Mathematics, Tata McGraw Hill, 2006.
2. Glyn James: Advanced Modern Engineering Mathematics, 3rd Edition, Pearson Education, 2003.

GRAPH THEORY AND COMBINATORICS
(Common to CSE & ISE)

Subject Code: 10CS42
Hours/Week : 04
Total Hours : 52

I.A. Marks : 25
Exam Hours: 03
Exam Marks: 100

PART – A

UNIT - 1 **7 Hours**
Introduction to Graph Theory: Definitions and Examples, Subgraphs, Complements, and Graph Isomorphism, Vertex Degree, Euler Trails and Circuits

UNIT – 2 **6 Hours**
Introduction to Graph Theory contd.: Planar Graphs, Hamilton Paths and Cycles, Graph Colouring, and Chromatic Polynomials

UNIT - 3 **6 Hours**
Trees: Definitions, Properties, and Examples, Routed Trees, Trees and Sorting, Weighted Trees and Prefix Codes

UNIT - 4 **7 Hours**
Optimization and Matching: Dijkstra's Shortest Path Algorithm, Minimal Spanning Trees – The algorithms of Kruskal and Prim, Transport Networks – Max-flow, Min-cut Theorem, Matching Theory

PART – B

UNIT - 5 **6 Hours**
Fundamental Principles of Counting: The Rules of Sum and Product, Permutations, Combinations – The Binomial Theorem, Combinations with Repetition, The Catalan Numbers

UNIT - 6 **6 Hours**
The Principle of Inclusion and Exclusion: The Principle of Inclusion and Exclusion, Generalizations of the Principle, Derangements – Nothing is in its Right Place, Rook Polynomials

UNIT - 7**7 Hours**

Generating Functions: Introductory Examples, Definition and Examples – Calculational Techniques, Partitions of Integers, the Exponential Generating Function, the Summation Operator

UNIT - 8**7 Hours**

Recurrence Relations: First Order Linear Recurrence Relation, The Second Order Linear Homogeneous Recurrence Relation with Constant Coefficients, The Non-homogeneous Recurrence Relation, The Method of Generating Functions

Text Book:

1. Ralph P. Grimaldi: Discrete and Combinatorial Mathematics, 5th Edition, Pearson Education, 2004.
(Chapter 11, Chapter 12.1 to 12.4, Chapter 13, Chapter 1, Chapter 8.1 to 8.4, Chapter 9 Chapter 10.1 to 10.4).

Reference Books:

1. D.S. Chandrasekharaiah: Graph Theory and Combinatorics, Prism, 2005.
2. Chartrand Zhang: Introduction to Graph Theory, TMH, 2006.
3. Richard A. Brualdi: Introductory Combinatorics, 4th Edition, Pearson Education, 2004.
4. Geir Agnarsson & Raymond Geenlaw: Graph Theory, Pearson Education, 2007.

DESIGN AND ANALYSIS OF ALGORITHMS
(Common to CSE & ISE)

Subject Code: 10CS43	I.A. Marks : 25
Hours/Week : 04	Exam Hours: 03
Total Hours : 52	Exam Marks: 100

PART – A**UNIT – 1****7 Hours**

INTRODUCTION: Notion of Algorithm, Review of Asymptotic Notations, Mathematical Analysis of Non-Recursive and Recursive Algorithms
Brute Force Approaches: Introduction, Selection Sort and Bubble Sort, Sequential Search and Brute Force String Matching.

UNIT - 2**6 Hours**

DIVIDE AND CONQUER: Divide and Conquer: General Method, Defective Chess Board, Binary Search, Merge Sort, Quick Sort and its performance.

UNIT - 3 **7 Hours**

THE GREEDY METHOD: The General Method, Knapsack Problem, Job Sequencing with Deadlines, Minimum-Cost Spanning Trees: Prim's Algorithm, Kruskal's Algorithm; Single Source Shortest Paths.

UNIT - 4 **6 Hours**

DYNAMIC PROGRAMMING: The General Method, Warshall's Algorithm, Floyd's Algorithm for the All-Pairs Shortest Paths Problem, Single-Source Shortest Paths: General Weights, 0/1 Knapsack, The Traveling Salesperson problem.

PART – B

UNIT - 5 **7 Hours**

DECREASE-AND-CONQUER APPROACHES, SPACE-TIME TRADEOFFS: Decrease-and-Conquer Approaches: Introduction, Insertion Sort, Depth First Search and Breadth First Search, Topological Sorting
Space-Time Tradeoffs: Introduction, Sorting by Counting, Input Enhancement in String Matching.

UNIT – 6 **7 Hours**

LIMITATIONS OF ALGORITHMIC POWER AND COPING WITH THEM: Lower-Bound Arguments, Decision Trees, P, NP, and NP-Complete Problems, Challenges of Numerical Algorithms.

UNIT - 7 **6 Hours**

COPING WITH LIMITATIONS OF ALGORITHMIC POWER:
Backtracking: n - Queens problem, Hamiltonian Circuit Problem, Subset – Sum Problem.
Branch-and-Bound: Assignment Problem, Knapsack Problem, Traveling Salesperson Problem.
Approximation Algorithms for NP-Hard Problems – Traveling Salesperson Problem, Knapsack Problem

UNIT – 8 **6 Hours**

PRAM ALGORITHMS: Introduction, Computational Model, Parallel Algorithms for Prefix Computation, List Ranking, and Graph Problems,

Text Books:

1. Anany Levitin: Introduction to The Design & Analysis of Algorithms, 2nd Edition, Pearson Education, 2007.
(Listed topics only from the Chapters 1, 2, 3, 5, 7, 8, 10, 11).
2. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran: Fundamentals of Computer Algorithms, 2nd Edition, Universities Press, 2007.
(Listed topics only from the Chapters 3, 4, 5, 13)

Reference Books:

1. Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein: Introduction to Algorithms, 3rd Edition, PHI, 2010.
2. R.C.T. Lee, S.S. Tseng, R.C. Chang & Y.T.Tsai: Introduction to the Design and Analysis of Algorithms A Strategic Approach, Tata McGraw Hill, 2005.

UNIX AND SHELL PROGRAMMING
(Common to CSE & ISE)

Subject Code: 10CS44	I.A. Marks : 25
Hours/Week : 04	Exam Hours: 03
Total Hours : 52	Exam Marks: 100

PART – A

UNIT – 1	6 Hours
The Unix Operating System, The UNIX architecture and Command Usage, The File System	
UNIT - 2	6 Hours
Basic File Attributes, the vi Editor	
UNIT – 3	7 Hours
The Shell, The Process, Customizing the environment	
UNIT - 4	7 Hours
More file attributes, Simple filters	

PART – B

UNIT – 5	6 Hours
Filters using regular expressions,	
UNIT – 6	6 Hours
Essential Shell Programming	

UNIT - 7 **7 Hours**
awk – An Advanced Filter

UNIT - 8 **7 Hours**
perl - The Master Manipulator

Text Book:

1. Sumitabha Das: UNIX – Concepts and Applications, 4th Edition, Tata McGraw Hill, 2006.
(Chapters 1.2, 2, 4, 6, 7, 8, 9, 10, 11, 12, 13, 14, 18, 19)

Reference Books:

1. Behrouz A. Forouzan and Richard F. Gilberg: UNIX and Shell Programming, Cengage Learning, 2005.
2. M.G. Venkateshmurthy: UNIX & Shell Programming, Pearson Education, 2005.

MICROPROCESSORS
(Common to CSE & ISE)

Subject Code: 10CS45

Hours/Week : 04

Total Hours : 52

I.A. Marks : 25

Exam Hours: 03

Exam Marks: 100

PART A

UNIT – I **7 Hours**
Introduction, Microprocessor Architecture – 1: A Historical Background, The Microprocessor-Based Personal Computer Systems.
The Microprocessor and its Architecture: Internal Microprocessor Architecture, Real Mode Memory Addressing.

UNIT – 2 **7 Hours**
Microprocessor Architecture – 2, Addressing Modes: Introduction to Protected Mode Memory Addressing, Memory Paging, Flat Mode Memory Addressing Modes: Data Addressing Modes, Program Memory Addressing Modes, Stack Memory Addressing Modes

UNIT – 3 **6 Hours**
Programming – 1: Data Movement Instructions: MOV Revisited, PUSH/POP, Load-Effective Address, String Data Transfers, Miscellaneous Data Transfer Instructions, Segment Override Prefix, Assembler Details.
Arithmetic and Logic Instructions: Addition, Subtraction and Comparison, Multiplication and Division.

UNIT - 4**6 Hours**

Programming – 2: Arithmetic and Logic Instructions (continued): BCD and ASCII Arithmetic, Basic Logic Instructions, Shift and Rotate, String Comparisons.

Program Control Instructions: The Jump Group, Controlling the Flow of the Program, Procedures, Introduction to Interrupts, Machine Control and Miscellaneous Instructions.

PART B**UNIT - 5****6 Hours**

Programming – 3: Combining Assembly Language with C/C++: Using Assembly Language with C/C++ for 16-Bit DOS Applications and 32-Bit Applications

Modular Programming, Using the Keyboard and Video Display, Data Conversions, Example Programs

UNIT - 6**7 Hours**

Hardware Specifications, Memory Interface – 1: Pin-Outs and the Pin Functions, Clock Generator, Bus Buffering and Latching, Bus Timings, Ready and Wait State, Minimum versus Maximum Mode.

Memory Interfacing: Memory Devices

UNIT – 7**6 Hours**

Memory Interface – 2, I/O Interface – 1: Memory Interfacing (continued): Address Decoding, 8088 Memory Interface, 8086 Memory Interface.

Basic I/O Interface: Introduction to I/O Interface, I/O Port Address Decoding.

UNIT 8**7 Hours**

I/O Interface – 2, Interrupts, and DMA: I/O Interface (continued): The Programmable Peripheral Interface 82C55, Programmable Interval Timer 8254.

Interrupts: Basic Interrupt Processing, Hardware Interrupts: INTR and INTA/; Direct Memory Access: Basic DMA Operation and Definition.

Text Book:

1. Barry B Brey: The Intel Microprocessors, 8th Edition, Pearson Education, 2009.
(Listed topics only from the Chapters 1 to 13)

Reference Books:

1. Douglas V. Hall: Microprocessors and Interfacing, Revised 2nd Edition, TMH, 2006.

2. K. Udaya Kumar & B.S. Umashankar : Advanced Microprocessors & IBM-PC Assembly Language Programming, TMH 2003.
3. James L. Antonakos: The Intel Microprocessor Family: Hardware and Software Principles and Applications, Cengage Learning, 2007.

COMPUTER ORGANIZATION
(Common to CSE & ISE)

Subject Code: 10CS46
Hours/Week : 04
Total Hours : 52

I.A. Marks : 25
Exam Hours: 03
Exam Marks: 100

PART – A

UNIT - 1 **6 Hours**

Basic Structure of Computers: Computer Types, Functional Units, Basic Operational Concepts, Bus Structures, Performance – Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement, Historical Perspective

Machine Instructions and Programs: Numbers, Arithmetic Operations and Characters, Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing,

UNIT - 2 **7 Hours**

Machine Instructions and Programs contd.: Addressing Modes, Assembly Language, Basic Input and Output Operations, Stacks and Queues, Subroutines, Additional Instructions, Encoding of Machine Instructions

UNIT - 3 **6 Hours**

Input/Output Organization: Accessing I/O Devices, Interrupts – Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Controlling Device Requests, Exceptions, Direct Memory Access, Buses

UNIT - 4 **7 Hours**

Input/Output Organization contd.: Interface Circuits, Standard I/O Interfaces – PCI Bus, SCSI Bus, USB

PART – B

UNIT - 5 **7 Hours**

Memory System: Basic Concepts, Semiconductor RAM Memories, Read Only Memories, Speed, Size, and Cost, Cache Memories – Mapping Functions, Replacement Algorithms, Performance Considerations, Virtual Memories, Secondary Storage

UNIT - 6 **7 Hours**
Arithmetic: Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed Operand Multiplication, Fast Multiplication, Integer Division, Floating-point Numbers and Operations

UNIT - 7 **6 Hours**
Basic Processing Unit: Some Fundamental Concepts, Execution of a Complete Instruction, Multiple Bus Organization, Hard-wired Control, Microprogrammed Control

UNIT - 8 **6 Hours**
Multicores, Multiprocessors, and Clusters: Performance, The Power Wall, The Switch from Uniprocessors to Multiprocessors, Amdahl's Law, Shared Memory Multiprocessors, Clusters and other Message Passing Multiprocessors, Hardware Multithreading, SISD, IMD, SIMD, SPMD, and Vector.

Text Books:

1. Carl Hamacher, Zvonko Vranesic, Safwat Zaky: Computer Organization, 5th Edition, Tata McGraw Hill, 2002.
(Listed topics only from Chapters 1, 2, 4, 5, 6, 7)
2. David A. Patterson, John L. Hennessy: Computer Organization and Design – The Hardware / Software Interface ARM Edition, 4th Edition, Elsevier, 2009.
(Listed topics only)

Reference Books:

1. William Stallings: Computer Organization & Architecture, 7th Edition, PHI, 2006.
2. Vincent P. Heuring & Harry F. Jordan: Computer Systems Design and Architecture, 2nd Edition, Pearson Education, 2004.

DESIGN AND ANALYSIS OF ALGORITHMS LABORATORY
(Common to CSE & ISE)

Subject Code: 10CSL47	I.A. Marks : 25
Hours/Week : 03	Exam Hours: 03
Total Hours : 42	Exam Marks: 50

Design, develop and implement the specified algorithms for the following problems using C/C++ Language in LINUX / Windows environment.

1. Sort a given set of elements using the Quicksort method and determine the time required to sort the elements. Repeat the experiment for different values of n , the number of elements in the list to be sorted and plot a graph of the time taken versus n .
The elements can be read from a file or can be generated using the random number generator.
2. Using OpenMP, implement a parallelized Merge Sort algorithm to sort a given set of elements and determine the time required to sort the elements. Repeat the experiment for different values of n , the number of elements in the list to be sorted and plot a graph of the time taken versus n . The elements can be read from a file or can be generated using the random number generator.
3.
 - a. Obtain the Topological ordering of vertices in a given digraph.
 - b. Compute the transitive closure of a given directed graph using Warshall's algorithm.
4. Implement 0/1 Knapsack problem using Dynamic Programming.
5. From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.
6. Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm.
7.
 - a. Print all the nodes reachable from a given starting node in a digraph using BFS method.
 - b. Check whether a given graph is connected or not using DFS method.
8. Find a subset of a given set $S = \{s_1, s_2, \dots, s_n\}$ of n positive integers whose sum is equal to a given positive integer d . For example, if $S = \{1, 2, 5, 6, 8\}$ and $d = 9$ there are two solutions $\{1, 2, 6\}$ and $\{1, 8\}$. A suitable message is to be displayed if the given problem instance doesn't have a solution.
9. Implement any scheme to find the optimal solution for the Traveling Salesperson problem and then solve the same problem instance using any approximation algorithm and determine the error in the approximation.
10. Find Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm.

11. Implement All-Pairs Shortest Paths Problem using Floyd's algorithm. Parallelize this algorithm, implement it using OpenMP and determine the speed-up achieved.
12. Implement N Queen's problem using Back Tracking.

Note: In the examination *each* student picks one question from the lot of *all* 12 questions.

MICROPROCESSORS LABORATORY
(Common to CSE & ISE)

Subject Code : 10CSL48	I.A. Marks : 25
Hours/Week : 03	Exam Hours: 03
Total Hours : 42	Exam Marks: 50

Notes:

- **Develop and execute the following programs using 8086 Assembly Language. Any suitable assembler like MASM, TASM etc may be used.**
 - **Program should have suitable comments.**
 - **The board layout and the circuit diagram of the interface are to be provided to the student during the examination.**
1. a) Search a key element in a list of 'n' 16-bit numbers using the Binary search algorithm.
b) Read the status of eight input bits from the Logic Controller Interface and display 'FF' if it is the parity of the input read is even; otherwise display 00.
 2. a) Write two ALP modules stored in two different files; one module is to read a character from the keyboard and the other one is to display a character. Use the above two modules to read a string of characters from the keyboard terminated by the carriage return and print the string on the display in the next line.
b) Implement a BCD Up-Down Counter on the Logic Controller Interface.

3. a) Sort a given set of 'n' numbers in ascending order using the Bubble Sort algorithm.
b) Read the status of two 8-bit inputs (X & Y) from the Logic Controller Interface and display $X*Y$.
4. a) Read an alphanumeric character and display its equivalent ASCII code at the center of the screen.
b) Display messages FIRE and HELP alternately with flickering effects on a 7-segment display interface for a suitable period of time. Ensure a flashing rate that makes it easy to read both the messages (Examiner does not specify these delay values nor is it necessary for the student to compute these values).
5. a) Reverse a given string and check whether it is a palindrome or not.
b) Assume any suitable message of 12 characters length and display it in the rolling fashion on a 7-segment display interface for a suitable period of time. Ensure a flashing rate that makes it easy to read both the messages. (Examiner does not specify these delay values nor is it necessary for the student to compute these values).
6. a) Read two strings, store them in locations STR1 and STR2. Check whether they are equal or not and display appropriate messages. Also display the length of the stored strings.
b) Convert a 16-bit binary value (assumed to be an unsigned integer) to BCD and display it from left to right and right to left for specified number of times on a 7-segment display interface.
7. a) Read your name from the keyboard and display it at a specified location on the screen after the message "**What is your name?**" You must clear the entire screen before display.
b) Scan a 8 x 3 keypad for key closure and to store the code of the key pressed in a memory location or display on screen. Also display row and column numbers of the key pressed.
8. a) Compute nCr using recursive procedure. Assume that 'n' and 'r' are non-negative integers.
b) Drive a Stepper Motor interface to rotate the motor in specified direction (clockwise or counter-clockwise) by N steps (Direction and N are specified by the examiner). Introduce suitable delay between successive steps. (Any arbitrary value for the delay may be assumed by the student).

9. a) Read the current time from the system and display it in the standard format on the screen.
b) Generate the Sine Wave using DAC interface (The output of the DAC is to be displayed on the CRO).
10. a) Write a program to simulate a Decimal Up-counter to display 00-99.
b) Generate a Half Rectified Sine wave form using the DAC interface. (The output of the DAC is to be displayed on the CRO).
11. a) Read a pair of input co-ordinates in BCD and move the cursor to the specified location on the screen.
b) Generate a Fully Rectified Sine waveform using the DAC interface. (The output of the DAC is to be displayed on the CRO).
12. a) Write a program to create a file (input file) and to delete an existing file.
b) Drive an elevator interface in the following way:
 - i. Initially the elevator should be in the ground floor, with all requests in OFF state.
 - ii. When a request is made from a floor, the elevator should move to that floor, wait there for a couple of seconds (approximately), and then come down to ground floor and stop. If some requests occur during going up or coming down they should be ignored.

Note: In the examination *each* student picks one question from the lot of *all* 12 questions.

V SEMESTER

SOFTWARE ENGINEERING

Subject Code: 10IS51

Hours/Week : 04

Total Hours : 52

I.A. Marks : 25

Exam Hours: 03

Exam Marks: 100

PART – A

UNIT – 1

6 Hours

Overview: Introduction: FAQ's about software engineering, Professional and ethical responsibility.

Socio-Technical systems: Emergent system properties; Systems engineering; Organizations, people and computer systems; Legacy systems.

UNIT – 2

6 Hours

Critical Systems, Software Processes: Critical Systems: A simple safety-critical system; System dependability; Availability and reliability.

Software Processes: Models, Process iteration, Process activities; The Rational Unified Process; Computer Aided Software Engineering.

UNIT – 3

7 Hours

Requirements: Software Requirements: Functional and Non-functional requirements; User requirements; System requirements; Interface specification; The software requirements document.

Requirements Engineering Processes: Feasibility studies; Requirements elicitation and analysis; Requirements validation; Requirements management.

UNIT – 4

7 Hours

System models, Project Management: System Models: Context models; Behavioral models; Data models; Object models; Structured methods.

Project Management: Management activities; Project planning; Project scheduling; Risk management

PART - B

UNIT – 5

7 Hours

Software Design: Architectural Design: Architectural design decisions; System organization; Modular decomposition styles; Control styles.

Object-Oriented design: Objects and Object Classes; An Object-Oriented design process; Design evolution.

UNIT – 6

6 Hours

Development: Rapid Software Development: Agile methods; Extreme programming; Rapid application development.
Software Evolution: Program evolution dynamics; Software maintenance; Evolution processes; Legacy system evolution.

UNIT – 7

7 Hours

Verification and Validation: Verification and Validation: Planning; Software inspections; Automated static analysis; Verification and formal methods.
Software testing: System testing; Component testing; Test case design; Test automation.

UNIT – 8

6 Hours

Management: Managing People: Selecting staff; Motivating people; Managing people; The People Capability Maturity Model.
Software Cost Estimation: Productivity; Estimation techniques; Algorithmic cost modeling, Project duration and staffing.

Text Books:

1. Ian Sommerville: Software Engineering, 8th Edition, Pearson Education, 2007.
(Chapters:- 1, 2, 3, 4, 5, 6, 7, 8, 11, 14, 17, 21, 22, 23, 25, 26)

Reference Books:

1. Roger.S.Pressman: Software Engineering-A Practitioners approach, 7th Edition, Tata McGraw Hill, 2007.
2. Pankaj Jalote: An Integrated Approach to Software Engineering, Wiley India, 2009.

SYSTEM SOFTWARE

Subject Code: 10CS52

I.A. Marks : 25

Hours/Week : 04

Exam Hours: 03

Total Hours : 52

Exam Marks: 100

PART – A

UNIT – 1

6 Hours

Machine Architecture: Introduction, System Software and Machine Architecture, Simplified Instructional Computer (SIC) - SIC Machine Architecture, SIC/XE Machine Architecture, SIC Programming Examples.

UNIT – 2

6 Hours

Assemblers -1: Basic Assembler Function - A Simple SIC Assembler, Assembler Algorithm and Data Structures, Machine Dependent Assembler Features - Instruction Formats & Addressing Modes, Program Relocation.

UNIT – 3

6 Hours

Assemblers -2: Machine Independent Assembler Features – Literals, Symbol-Definition Statements, Expression, Program Blocks, Control Sections and Programming Linking, Assembler Design Operations - One-Pass Assembler, Multi-Pass Assembler, Implementation Examples - MASM Assembler.

UNIT – 4

8 Hours

Loaders and Linkers: Basic Loader Functions - Design of an Absolute Loader, A Simple Bootstrap Loader, Machine-Dependent Loader Features – Relocation, Program Linking, Algorithm and Data Structures for a Linking Loader; Machine-Independent Loader Features - Automatic Library Search, Loader Options, Loader Design Options - Linkage Editor, Dynamic Linkage, Bootstrap Loaders, Implementation Examples - MS-DOS Linker.

PART – B

UNIT – 5

6 Hours

Editors and Debugging Systems: Text Editors - Overview of Editing Process, User Interface, Editor Structure, Interactive Debugging Systems - Debugging Functions and Capabilities, Relationship With Other Parts Of The System, User-Interface Criteria

UNIT – 6

8 Hours

Macro Processor: Basic Macro Processor Functions - Macro Definitions and Expansion, Macro Processor Algorithm and Data Structures, Machine-Independent Macro Processor Features - Concatenation of Macro Parameters, Generation of Unique Labels, Conditional Macro Expansion, Keyword Macro Parameters, Macro Processor Design Options - Recursive Macro Expansion, General-Purpose Macro Processors, Macro Processing Within Language Translators, Implementation Examples - MASM Macro Processor, ANSI C Macro Processor.

UNIT – 7

6 Hours

Lex and Yacc – 1: Lex and Yacc - The Simplest Lex Program, Recognizing Words With LEX, Symbol Tables, Grammars, Parser-Lexer Communication, The Parts of Speech Lexer, A YACC Parser, The Rules Section, Running

LEX and YACC, LEX and Hand- Written Lexers, Using LEX - Regular Expression, Examples of Regular Expressions, A Word Counting Program, Parsing a Command Line.

UNIT – 8

6 Hours

Lex and Yacc - 2: Using YACC – Grammars, Recursive Rules, Shift/Reduce Parsing, What YACC Cannot Parse, A YACC Parser - The Definition Section, The Rules Section, Symbol Values and Actions, The LEXER, Compiling and Running a Simple Parser, Arithmetic Expressions and Ambiguity, Variables and Typed Tokens.

Text Books:

1. Leland.L.Beck: System Software, 3rd Edition, Pearson Education, 1997.
(Chapters 1.1 to 1.3, 2 (except 2.5.2 and 2.5.3), 3 (except 3.5.2 and 3.5.3), 4 (except 4.4.3))
2. John.R.Levine, Tony Mason and Doug Brown: Lex and Yacc, O'Reilly, SPD, 1998.
(Chapters 1, 2 (Page 2-42), 3 (Page 51-65))

Reference Books:

1. D.M.Dhamdhare: System Programming and Operating Systems, 2nd Edition, Tata McGraw - Hill, 1999.

OPERATING SYSTEMS

Subject Code: 10CS53

I.A. Marks : 25

Hours/Week : 04

Exam Hours: 03

Total Hours : 52

Exam Marks: 100

PART – A

UNIT – 1

6 Hours

Introduction to Operating Systems, System structures: What operating systems do; Computer System organization; Computer System architecture; Operating System structure; Operating System operations; Process management; Memory management; Storage management; Protection and security; Distributed system; Special-purpose systems; Computing environments. Operating System Services; User - Operating System interface; System calls; Types of system calls; System programs; Operating System design and implementation; Operating System structure; Virtual machines; Operating System generation; System boot.

UNIT – 2 **7 Hours**
Process Management: Process concept; Process scheduling; Operations on processes; Inter-process communication. Multi-Threaded Programming: Overview; Multithreading models; Thread Libraries; Threading issues. Process Scheduling: Basic concepts; Scheduling criteria; Scheduling algorithms; Multiple-Processor scheduling; Thread scheduling.

UNIT – 3 **7 Hours**
Process Synchronization : Synchronization: The Critical section problem; Peterson’s solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors.

UNIT – 4 **6 Hours**
Deadlocks: Deadlocks: System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock.

PART – B

UNIT – 5 **7 Hours**
Memory Management: Memory Management Strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. Virtual Memory Management: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing.

UNIT – 6 **7 Hours**
File System, Implementation of File System: File System: File concept; Access methods; Directory structure; File system mounting; File sharing; Protection. Implementing File System: File system structure; File system implementation; Directory implementation; Allocation methods; Free space management

UNIT – 7 **6 Hours**
Secondary Storage Structures, Protection : Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Disk management; Swap space management. Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix, Implementation of access matrix, Access control, Revocation of access rights, Capability-Based systems.

UNIT – 8 **6 Hours**
Case Study: The Linux Operating System: Linux history; Design principles; Kernel modules; Process management; Scheduling; Memory management; File systems, Input and output; Inter-process communication.

Text Books:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 8th edition, Wiley India, 2009.
(Listed topics only from Chapters 1 to 12, 17, 21)

Reference Books:

1. D.M Dhamdhare: Operating systems - A concept based Approach, 2nd Edition, Tata McGraw- Hill, 2002.
2. P.C.P. Bhatt: Introduction to Operating Systems: Concepts and Practice, 2nd Edition, PHI, 2008.
3. Harvey M Deital: Operating systems, 3rd Edition, Pearson Education, 1990.

DATABASE MANAGEMENT SYSTEMS**Subject Code: 10CS54****Hours/Week : 04****Total Hours : 52****I.A. Marks : 25****Exam Hours: 03****Exam Marks: 100****PART - A****UNIT – 1****6 Hours**

Introduction: Introduction; An example; Characteristics of Database approach; Actors on the screen; Workers behind the scene; Advantages of using DBMS approach; A brief history of database applications; when not to use a DBMS.

Data models, schemas and instances; Three-schema architecture and data independence; Database languages and interfaces; The database system environment; Centralized and client-server architectures; Classification of Database Management systems.

UNIT – 2**6 Hours**

Entity-Relationship Model: Using High-Level Conceptual Data Models for Database Design; An Example Database Application; Entity Types, Entity Sets, Attributes and Keys; Relationship types, Relationship Sets, Roles and Structural Constraints; Weak Entity Types; Refining the ER Design; ER Diagrams, Naming Conventions and Design Issues; Relationship types of degree higher than two.

UNIT – 3**8 Hours**

Relational Model and Relational Algebra : Relational Model Concepts; Relational Model Constraints and Relational Database Schemas; Update

Operations, Transactions and dealing with constraint violations; Unary Relational Operations: SELECT and PROJECT; Relational Algebra Operations from Set Theory; Binary Relational Operations : JOIN and DIVISION; Additional Relational Operations; Examples of Queries in Relational Algebra; Relational Database Design Using ER- to-Relational Mapping.

UNIT – 4

6 Hours

SQL – 1: SQL Data Definition and Data Types; Specifying basic constraints in SQL; Schema change statements in SQL; Basic queries in SQL; More complex SQL Queries.

PART - B

UNIT – 5

6 Hours

SQL – 2: Insert, Delete and Update statements in SQL; Specifying constraints as Assertion and Trigger; Views (Virtual Tables) in SQL; Additional features of SQL; Database programming issues and techniques; Embedded SQL, Dynamic SQL; Database stored procedures and SQL / PSM.

UNIT – 6

6 Hours

Database Design – 1: Informal Design Guidelines for Relation Schemas; Functional Dependencies; Normal Forms Based on Primary Keys; General Definitions of Second and Third Normal Forms; Boyce-Codd Normal Form

UNIT – 7

6 Hours

Database Design -2: Properties of Relational Decompositions; Algorithms for Relational Database Schema Design; Multivalued Dependencies and Fourth Normal Form; Join Dependencies and Fifth Normal Form; Inclusion Dependencies; Other Dependencies and Normal Forms

UNIT – 8

8 Hours

Transaction Management: The ACID Properties; Transactions and Schedules; Concurrent Execution of Transactions; Lock- Based Concurrency Control; Performance of locking; Transaction support in SQL; Introduction to crash recovery; 2PL, Serializability and Recoverability; Lock Management; Introduction to ARIES; The log; Other recovery-related structures; The write-ahead log protocol; Checkpointing; Recovering from a System Crash; Media Recovery; Other approaches and interaction with concurrency control.

Text Books:

1. Elmasri and Navathe: Fundamentals of Database Systems, 5th Edition, Pearson Education, 2007.

(Chapters 1, 2, 3 except 3.8, 5, 6.1 to 6.5, 7.1, 8, 9.1, 9.2 except SQLJ, 9.4, 10)

2. Raghu Ramakrishnan and Johannes Gehrke: Database Management Systems, 3rd Edition, McGraw-Hill, 2003. (Chapters 16, 17.1, 17.2, 18)

Reference Books:

1. Silberschatz, Korth and Sudharshan: Data base System Concepts, 6th Edition, Mc-GrawHill, 2010.
2. C.J. Date, A. Kannan, S. Swamynatham: An Introduction to Database Systems, 8th Edition, Pearson Education, 2006.

COMPUTER NETWORKS - I

Subject Code: 10CS55

Hours/Week : 04

Total Hours : 52

I.A. Marks : 25

Exam Hours: 03

Exam Marks: 100

PART – A

UNIT - 1

7 Hours

Introduction: Data Communications, Networks, The Internet, Protocols & Standards, Layered Tasks, The OSI model, Layers in OSI model, TCP/IP Protocol suite, Addressing

UNIT- 2

7 Hours

Physical Layer-1: Analog & Digital Signals, Transmission Impairment, Data Rate limits, Performance, Digital-digital conversion (Only Line coding: Polar, Bipolar and Manchester coding), Analog-to-digital conversion (only PCM), Transmission Modes, Digital-to-analog conversion

UNIT- 3

6 Hours

Physical Layer-2 and Switching: Multiplexing, Spread Spectrum, Introduction to switching, Circuit Switched Networks, Datagram Networks, Virtual Circuit Networks

UNIT- 4

6 Hours

Data Link Layer-1: Error Detection & Correction: Introduction, Block coding, Linear block codes, Cyclic codes, Checksum.

PART - B

UNIT- 5 **6 Hours**
Data Link Layer-2: Framing, Flow and Error Control, Protocols, Noiseless Channels, Noisy channels, HDLC, PPP (Framing, Transition phases only)

UNIT- 6 **7 Hours**
Multiple Access & Ethernet: Random access, Controlled Access, Channelization, Ethernet: IEEE standards, Standard Ethernet, Changes in the standard, Fast Ethernet, Gigabit Ethernet

UNIT - 7 **6 Hours**
Wireless LANs and Cellular Networks: Introduction, IEEE 802.11, Bluetooth, Connecting devices, Cellular Telephony

UNIT - 8: **7 Hours**
Network Layer: Introduction, Logical addressing, IPv4 addresses, IPv6 addresses, Internetworking basics, IPv4, IPv6, Comparison of IPv4 and IPv6 Headers.

Text Books:

1. Behrouz A. Forouzan,: Data Communication and Networking, 4th Edition Tata McGraw-Hill, 2006.
(Chapters 1.1 to 1.4, 2.1 to 2.5, 3.1 To 3.6, 4.1 to 4.3, 5.1, 6.1, 6.2, 8.1 to 8.3, 10.1 to 10.5, 11.1 to 11.7, 12.1 to 12.3, 13.1 to 13.5, 14.1, 14.2, 15.1, 16.1, 19.1, 19.2, 20.1 to 20.3)

Reference Books:

1. Alberto Leon-Garcia and Indra Widjaja: Communication Networks - Fundamental Concepts and Key architectures, 2nd Edition Tata McGraw-Hill, 2004.
2. William Stallings: Data and Computer Communication, 8th Edition, Pearson Education, 2007.
3. Larry L. Peterson and Bruce S. Davie: Computer Networks – A Systems Approach, 4th Edition, Elsevier, 2007.
4. Nader F. Mir: Computer and Communication Networks, Pearson Education, 2007.

Extensions to the basic Turing Machines; Turing Machine and Computers.

UNIT – 8

6 Hours

Undecidability: A Language that is not recursively enumerable; An Undecidable problem that is RE; Post's Correspondence problem; Other undecidable problems.

Text Books:

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D.Ullman: Introduction to Automata Theory, Languages and Computation, 3rd Edition, Pearson Education, 2007.
(Chapters: 1.1, 1.5, 2.2 to 2.5, 3.1 to 3.3, 4, 5, 6, 7, 8.1 to 8.4, 8.6, 9.1, 9.2, 9.4.1, 9.5)

Reference Books:

1. K.L.P. Mishra: Theory of Computer Science, Automata, Languages, and Computation, 3rd Edition, PHI Learning, 2009.
2. Raymond Greenlaw, H.James Hoover: Fundamentals of the Theory of Computation, Principles and Practice, Elsevier, 1998.
3. John C Martin: Introduction to Languages and Automata Theory, 3rd Edition, Tata McGraw-Hill, 2007.
4. Thomas A. Sudkamp: An Introduction to the Theory of Computer Science, Languages and Machines, 3rd Edition, Pearson Education, 2006.

DATABASE APPLICATIONS LABORATORY

Subject Code: 10CSL57

I.A. Marks : 25

Hours/Week : 03

Exam Hours: 03

Total Hours : 42

Exam Marks: 50

1. Consider the following relations:

Student (*snum*: integer, *sname*: string, *major*: string, *level*: string, *age*: integer)

Class (*name*: string, *meets at*: string, *room*: string, *d*: integer)

Enrolled (*snum*: integer, *cname*: string)

Faculty (*fid*: integer, *fname*: string, *deptid*: integer)

The meaning of these relations is straightforward; for example, Enrolled has one record per student-class pair such that the student is enrolled in the class. Level is a two character code with 4 different values (example: Junior: JR etc)

Write the following queries in SQL. No duplicates should be printed in any of the answers.

- i. Find the names of all Juniors (level = JR) who are enrolled in a class taught by Prof. Harshith
- ii. Find the names of all classes that either meet in room R128 or have five or more Students enrolled.
- iii. Find the names of all students who are enrolled in two classes that meet at the same time.
- iv. Find the names of faculty members who teach in every room in which some class is taught.
- v. Find the names of faculty members for whom the combined enrollment of the courses that they teach is less than five.

2. The following relations keep track of airline flight information:

Flights (*no*: integer, *from*: string, *to*: string, *distance*: integer, *Departs*: time, *arrives*: time, *price*: real)

Aircraft (*aid*: integer, *aname*: string, *cruisingrange*: integer)

Certified (*eid*: integer, *aid*: integer)

Employees (*eid*: integer, *ename*: string, *salary*: integer)

Note that the Employees relation describes pilots and other kinds of employees as well; Every pilot is certified for some aircraft, and only pilots are certified to fly.

Write each of the following queries in SQL.

- i. Find the names of aircraft such that all pilots certified to operate them have salaries more than Rs.80, 000.
- ii. For each pilot who is certified for more than three aircrafts, find the *eid* and the maximum *cruisingrange* of the aircraft for which she or he is certified.
- iii. Find the names of pilots whose *salary* is less than the price of the cheapest route from Bengaluru to Frankfurt.
- iv. For all aircraft with *cruisingrange* over 1000 Kms, find the name of the aircraft and the average salary of all pilots certified for this aircraft.
- v. Find the names of pilots certified for some Boeing aircraft.
- vi. Find the *aids* of all aircraft that can be used on routes from Bengaluru to New Delhi.

3. Consider the following database of student enrollment in courses & books adopted for each course.

STUDENT (regno: string, name: string, major: string, bdate:date)

COURSE (course #:int, cname:string, dept:string)

ENROLL (regno:string, course#:int, sem:int, marks:int)

BOOK _ ADOPTION (course#:int, sem:int, book-ISBN:int)

TEXT (book-ISBN:int, book-title:string, publisher:string, author:string)

- i. Create the above tables by properly specifying the primary keys and the foreign keys.
 - ii. Enter at least five tuples for each relation.
 - iii. Demonstrate how you add a new text book to the database and make this book be adopted by some department.
 - iv. Produce a list of text books (include Course #, Book-ISBN, Book-title) in the alphabetical order for courses offered by the 'CS' department that use more than two books.
 - v. List any department that has *all* its adopted books published by a specific publisher.
 - vi. Generate suitable reports.
 - vii. Create suitable front end for querying and displaying the results.
4. The following tables are maintained by a book dealer.
- AUTHOR (author-id:int, name:string, city:string, country:string)
PUBLISHER (publisher-id:int, name:string, city:string, country:string)
CATALOG (book-id:int, title:string, author-id:int, publisher-id:int, category-id:int, year:int, price:int)
CATEGORY (category-id:int, description:string)
ORDER-DETAILS (order-no:int, book-id:int, quantity:int)
- i. Create the above tables by properly specifying the primary keys and the foreign keys.
 - ii. Enter at least five tuples for each relation.
 - iii. Give the details of the authors who have 2 or more books in the catalog and the price of the books is greater than the average price of the books in the catalog and the year of publication is after 2000.
 - iv. Find the author of the book which has maximum sales.
 - v. Demonstrate how you increase the price of books published by a specific publisher by 10%.
 - vi. Generate suitable reports.
 - vii. Create suitable front end for querying and displaying the results.
5. Consider the following database for a banking enterprise
- BRANCH(branch-name:string, branch-city:string, assets:real)
ACCOUNT(accno:int, branch-name:string, balance:real)
DEPOSITOR(customer-name:string, accno:int)
CUSTOMER(customer-name:string, customer-street:string, customer-city:string)
LOAN(loan-number:int, branch-name:string, amount:real)
BORROWER(customer-name:string, loan-number:int)

- i. Create the above tables by properly specifying the primary keys and the foreign keys
- ii. Enter at least five tuples for each relation
- iii. Find all the customers who have at least two accounts at the *Main* branch.
- iv. Find all the customers who have an account at *all* the branches located in a specific city.
- v. Demonstrate how you delete all account tuples at every branch located in a specific city.
- vi. Generate suitable reports.
- vii. Create suitable front end for querying and displaying the results.

Instructions:

1. The exercises are to be solved in an RDBMS environment like Oracle or DB2.
2. Suitable tuples have to be entered so that queries are executed correctly.
3. Front end may be created using either VB or VAJ or any other similar tool.
4. The student need not create the front end in the examination. The results of the queries may be displayed directly.
5. Relevant queries other than the ones listed along with the exercises may also be asked in the examination.
6. Questions must be asked based on lots.

**SYSTEM SOFTWARE & OPERATING SYSTEMS
LABORATORY**

Subject Code: 10CSL58	I.A. Marks : 25
Hours/Week : 03	Exam Hours: 03
Total Hours : 42	Exam Marks: 50

PART - A

LEX and YACC Programs:

Design, develop, and execute the following programs using LEX:

1. a) Program to count the number of characters, words, spaces and lines in a given input file.

- b) Program to count the numbers of comment lines in a given C program. Also eliminate them and copy the resulting program into separate file.
2. a) Program to recognize a valid arithmetic expression and to recognize the identifiers and operators present. Print them separately.
 - b) Program to recognize whether a given sentence is simple or compound.
 3. Program to recognize and count the number of identifiers in a given input file.

Design, develop, and execute the following programs using YACC:

4. a) Program to recognize a valid arithmetic expression that uses operators +, -, * and /.
 - b) Program to recognize a valid variable, which starts with a letter, followed by any number of letters or digits.
5. a) Program to evaluate an arithmetic expression involving operators +, -, * and /.
 - b) Program to recognize strings 'aab', 'abb', 'ab' and 'a' using the grammar $(a^n b^n, n \geq 0)$.
6. Program to recognize the grammar $(a^n b, n \geq 10)$.

PART B

UNIX Programming:

Design, develop, and execute the following programs:

7. a) Non-recursive shell script that accepts any number of arguments and prints them in the Reverse order, (For example, if the script is named rargs, then executing rargs A B C should produce C B A on the standard output).
 - b) C program that creates a child process to read commands from the standard input and execute them (a minimal implementation of a shell – like program). You can assume that no arguments will be passed to the commands to be executed.
8. a) Shell script that accepts two file names as arguments, checks if the permissions for these files are identical and if the permissions

are identical, outputs the common permissions, otherwise outputs each file name followed by its permissions.

- b) C program to create a file with 16 bytes of arbitrary data from the beginning and another 16 bytes of arbitrary data from an offset of 48. Display the file contents to demonstrate how the hole in file is handled.
9. a) Shell script that accepts file names specified as arguments and creates a shell script that contains this file as well as the code to recreate these files. Thus if the script generated by your script is executed, it would recreate the original files (This is same as the “bundle” script described by Brian W. Kernighan and Rob Pike in “The Unix Programming Environment”, Prentice – Hall India).
- b) C program to do the following: Using `fork()` create a child process. The child process prints its own process-id and id of its parent and then exits. The parent process waits for its child to finish (by executing the `wait()`) and prints its own process-id and the id of its child process and then exits.

Operating Systems:

10. Design, develop and execute a program in C / C++ to simulate the working of Shortest Remaining Time and Round-Robin Scheduling Algorithms. Experiment with different quantum sizes for the Round-Robin algorithm. In all cases, determine the average turn-around time. The input can be read from key board or from a file.
11. Using OpenMP, Design, develop and run a multi-threaded program to generate and print Fibonacci Series. One thread has to generate the numbers up to the specified limit and another thread has to print them. Ensure proper synchronization.
12. Design, develop and run a program to implement the Banker’s Algorithm. Demonstrate its working with different data values.

Instructions:

In the examination, a combination of one LEX and one YACC problem has to be asked from Part A for a total of 30 marks and one programming exercise from Part B has to be asked for a total of 20 marks.

VI SEMESTER

MANAGEMENT AND ENTREPRENEURSHIP (Common to All Branches)

Subject Code: 10AL61	I.A. Marks : 25
Hours/Week : 04	Exam Hours: 03
Total Hours : 52	Exam Marks: 100

UNIX SYSTEM PROGRAMMING

Subject Code: 10CS62	I.A. Marks : 25
Hours/Week : 04	Exam Hours: 03
Total Hours : 52	Exam Marks: 100

PART - A

UNIT – 1 **6 Hours**

Introduction: UNIX and ANSI Standards: The ANSI C Standard, The ANSI/ISO C++ Standards, Difference between ANSI C and C++, The POSIX Standards, The POSIX.1 FIPS Standard, The X/Open Standards.
UNIX and POSIX APIs: The POSIX APIs, The UNIX and POSIX Development Environment, API Common Characteristics.

UNIT – 2 **6 Hours**

UNIX Files: File Types, The UNIX and POSIX File System, The UNIX and POSIX File Attributes, Inodes in UNIX System V, Application Program Interface to Files, UNIX Kernel Support for Files, Relationship of C Stream Pointers and File Descriptors, Directory Files, Hard and Symbolic Links.

UNIT – 3 **7 Hours**

UNIX File APIs: General File APIs, File and Record Locking, Directory File APIs, Device File APIs, FIFO File APIs, Symbolic Link File APIs, General File Class, regfile Class for Regular Files, dirfile Class for Directory Files, FIFO File Class, Device File Class, Symbolic Link File Class, File Listing Program.

UNIT – 4 **7 Hours**

UNIX Processes: The Environment of a UNIX Process: Introduction, main function, Process Termination, Command-Line Arguments, Environment List, Memory Layout of a C Program, Shared Libraries, Memory Allocation,

Environment Variables, setjmp and longjmp Functions, getrlimit, setrlimit Functions, UNIX Kernel Support for Processes.

PART - B

UNIT – 5

7 Hours

Process Control : Introduction, Process Identifiers, fork, vfork, exit, wait, waitpid, wait3, wait4 Functions, Race Conditions, exec Functions, Changing User IDs and Group IDs, Interpreter Files, system Function, Process Accounting, User Identification, Process Times, I/O Redirection.

Process Relationships: Introduction, Terminal Logins, Network Logins, Process Groups, Sessions, Controlling Terminal, tcgetpgrp and tcsetpgrp Functions, Job Control, Shell Execution of Programs, Orphaned Process Groups.

UNIT – 6

7 Hours

Signals and Daemon Processes: Signals: The UNIX Kernel Support for Signals, signal, Signal Mask, sigaction, The SIGCHLD Signal and the waitpid Function, The sigsetjmp and siglongjmp Functions, Kill, Alarm, Interval Timers, POSIX.lb Timers.

Daemon Processes: Introduction, Daemon Characteristics, Coding Rules, Error Logging, Client-Server Model.

UNIT – 7

6 Hours

Interprocess Communication – 1: Overview of IPC Methods, Pipes, popen, pclose Functions, Coprocesses, FIFOs, System V IPC, Message Queues, Semaphores.

UNIT – 8

6 Hours

Interprocess Communication – 2: Shared Memory, Client-Server Properties, Stream Pipes, Passing File Descriptors, An Open Server-Version 1, Client-Server Connection Functions.

Text Books:

1. Terrence Chan: UNIX System Programming Using C++, Prentice Hall India, 1999.
(Chapters 1, 5, 6, 7, 8, 9, 10)
2. W. Richard Stevens: Advanced Programming in the UNIX Environment, 2nd Edition, Pearson Education, 2005.
(Chapters 7, 8, 9, 13, 14, 15)

Reference Books:

1. Marc J. Rochkind: Advanced UNIX Programming, 2nd Edition, Pearson Education, 2005.
2. Maurice J Bach: The Design of the UNIX Operating System, Pearson Education, 1987.

COMPUTER NETWORKS - II

Subject Code: 10CS64
Hours/Week : 04
Total Hours : 52

I.A. Marks : 25
Exam Hours: 03
Exam Marks: 100

PART - A

UNIT - 1

6 Hours

Packet Switching Networks - 1: Network services and internal network operation, Packet network topology, Routing in Packet networks, Shortest path routing: Bellman-Ford algorithm.

UNIT – 2

6 Hours

Packet Switching Networks – 2: Shortest path routing (continued), Traffic management at the Packet level, Traffic management at Flow level, Traffic management at flow aggregate level.

UNIT – 3

6 Hours

TCP/IP-1: TCP/IP architecture, The Internet Protocol, IPv6, UDP.

UNIT – 4

8 Hours

TCP/IP-2: TCP, Internet Routing Protocols, Multicast Routing, DHCP, NAT and Mobile IP.

PART – B

UNIT - 5

7 Hours

Applications, Network Management, Network Security: Application layer overview, Domain Name System (DNS), Remote Login Protocols, E-mail, File Transfer and FTP, World Wide Web and HTTP, Network management, Overview of network security, Overview of security methods, Secret-key encryption protocols, Public-key encryption protocols, Authentication, Authentication and digital signature, Firewalls.

UNIT – 6

6 Hours

QoS, VPNs, Tunneling, Overlay Networks: Overview of QoS, Integrated Services QoS, Differentiated services QoS, Virtual Private Networks, MPLS, Overlay networks.

UNIT - 7

7 Hours

Multimedia Networking: Overview of data compression, Digital voice and compression, JPEG, MPEG, Limits of compression with loss, Compression methods without loss, Overview of IP Telephony, VoIP signaling protocols, Real-Time Media Transport Protocols, Stream control Transmission Protocol (SCTP)

UNIT – 8

6 Hours

Mobile AdHoc Networks and Wireless Sensor Networks: Overview of Wireless Ad-Hoc networks, Routing in AdHOc Networks, Routing protocols for and Security of AdHoc networks, Sensor Networks and protocol structures, Communication Energy model, Clustering protocols, Routing protocols, ZigBee technology and 802.15.4.

Text Books:

1. Communication Networks – Fundamental Concepts & key architectures, Alberto Leon Garcia & Indra Widjaja, 2nd Edition, Tata McGraw-Hill, India
(7 - excluding 7.6, 8)
2. Computer & Communication Networks, Nadir F Mir, Pearson Education, India
(9, 10 excluding 10.7, 12.1 to 12.3, 16, 17.1 to 17.6, 18.1 to 18.3, 18.5, 19, 20)

Reference Books:

1. Behrouz A. Forouzan: Data Communications and Networking, 4th Edition, Tata McGraw-Hill, 2006.
2. William Stallings: Data and Computer Communication, 8th Edition, Pearson Education, 2007.
3. Larry L. Peterson and Bruce S. Davie: Computer Networks – A Systems Approach, 4th Edition, Elsevier, 2007.
4. Wayne Tomasi: Introduction to Data Communications and Networking, Pearson Education, 2005.

COMPUTER GRAPHICS AND VISUALIZATION

Subject Code: 10CS65
Hours/Week : 04
Total Hours : 52

I.A. Marks : 25
Exam Hours: 03
Exam Marks: 100

PART - A

UNIT – 1

7 Hours

Introduction: Applications of computer graphics; A graphics system; Images: Physical and synthetic; Imaging Systems; The synthetic camera model; The programmer's interface; Graphics architectures; Programmable Pipelines; Performance Characteristics
Graphics Programming: The Sierpinski gasket; Programming Two Dimensional Applications.

UNIT – 2

6 Hours

The OpenGL: The OpenGL API; Primitives and attributes; Color; Viewing; Control functions; The Gasket program; Polygons and recursion; The three-dimensional gasket; Plotting Implicit Functions

UNIT – 3**7 Hours**

Input and Interaction: Interaction; Input devices; Clients and Servers; Display Lists; Display Lists and Modeling; Programming Event Driven Input; Menus; Picking; A simple CAD program; Building Interactive Models; Animating Interactive Programs; Design of Interactive Programs; Logic Operations

UNIT – 4**6 Hours**

Geometric Objects and Transformations-I: Scalars, Points, and Vectors; Three-dimensional Primitives; Coordinate Systems and Frames; Modeling a Colored Cube; Affine Transformations; Rotation, Translation and Scaling;

PART - B**UNIT – 5****5 Hours**

Geometric Objects and Transformations-II: Geometric Objects and Transformations; Transformation in Homogeneous Coordinates; Concatenation of Transformations; OpenGL Transformation Matrices; Interfaces to three-dimensional applications; Quaternion's.

UNIT – 6**7 Hours**

Viewing: Classical and computer viewing; Viewing with a Computer; Positioning of the camera; Simple projections; Projections in OpenGL; Hidden-surface removal; Interactive Mesh Displays; Parallel-projection matrices; Perspective-projection matrices; Projections and Shadows.

UNIT – 7**6 Hours**

Lighting and Shading: Light and Matter; Light Sources; The Phong Lighting model; Computation of vectors; Polygonal Shading; Approximation of a sphere by recursive subdivisions; Light sources in OpenGL; Specification of materials in OpenGL; Shading of the sphere model; Global Illumination.

UNIT – 8**8 Hours**

Implementation: Basic Implementation Strategies; Four major tasks; Clipping; Line-segment clipping; Polygon clipping; Clipping of other primitives; Clipping in three dimensions; Rasterization; Bresenham's algorithm; Polygon Rasterization; Hidden-surface removal; Antialiasing; Display considerations.

Text Books:

1. Edward Angel: Interactive Computer Graphics A Top-Down Approach with OpenGL, 5th Edition, Pearson Education, 2008. (Chapters 1 to 7)

Reference Books:

1. Donald Hearn and Pauline Baker: Computer Graphics- OpenGL Version, 3rd Edition, Pearson Education, 2004.

2. F.S. Hill Jr.: Computer Graphics Using OpenGL, 3rd Edition, PHI, 2009.
3. James D Foley, Andries Van Dam, Steven K Feiner, John F Hughes, Computer Graphics, Pearson Education 1997.

OPERATIONS RESEARCH

Subject Code: 10CS661
Hours/Week : 04
Total Hours : 52

I.A. Marks : 25
Exam Hours: 03
Exam Marks: 100

PART - A

UNIT – 1 **6 Hours**

Introduction, Linear Programming – 1: Introduction: The origin, nature and impact of OR; Defining the problem and gathering data; Formulating a mathematical model; Deriving solutions from the model; Testing the model; Preparing to apply the model; Implementation .

Introduction to Linear Programming: Prototype example; The linear programming (LP) model.

UNIT – 2 **7 Hours**

LP – 2, Simplex Method – 1: Assumptions of LP; Additional examples. The essence of the simplex method; Setting up the simplex method; Algebra of the simplex method; the simplex method in tabular form; Tie breaking in the simplex method

UNIT – 3 **6 Hours**

Simplex Method – 2: Adapting to other model forms; Post optimality analysis; Computer implementation
 Foundation of the simplex method.

UNIT – 4 **7 Hours**

Simplex Method – 2, Duality Theory: The revised simplex method, a fundamental insight.

The essence of duality theory; Economic interpretation of duality, Primal dual relationship; Adapting to other primal forms

PART - B

UNIT – 5 **7 Hours**

Duality Theory and Sensitivity Analysis, Other Algorithms for LP : The role of duality in sensitive analysis; The essence of sensitivity analysis;

Applying sensitivity analysis. The dual simplex method; Parametric linear programming; The upper bound technique.

UNIT – 6

7 Hours

Transportation and Assignment Problems: The transportation problem; A streamlined simplex method for the transportation problem; The assignment problem; A special algorithm for the assignment problem.

UNIT – 7

6 Hours

Game Theory, Decision Analysis: Game Theory: The formulation of two persons, zero sum games; Solving simple games- a prototype example; Games with mixed strategies; Graphical solution procedure; Solving by linear programming, Extensions.

Decision Analysis: A prototype example; Decision making without experimentation; Decision making with experimentation; Decision trees.

UNIT – 8

6 Hours

Metaheuristics: The nature of Metaheuristics, Tabu Search, Simulated Annealing, Genetic Algorithms.

Text Books:

1. Frederick S. Hillier and Gerald J. Lieberman: Introduction to Operations Research: Concepts and Cases, 8th Edition, Tata McGraw Hill, 2005.
(Chapters: 1, 2, 3.1 to 3.4, 4.1 to 4.8, 5, 6.1 to 6.7, 7.1 to 7.3, 8, 13, 14, 15.1 to 15.4)

Reference Books:

1. Wayne L. Winston: Operations Research Applications and Algorithms, 4th Edition, Cengage Learning, 2003.
2. Hamdy A Taha: Operations Research: An Introduction, 8th Edition, Pearson Education, 2007.

COMPUTER GRAPHICS AND VISUALIZATION LABORATORY

Subject Code: 10CSL67
Hours/Week : 03
Total Hours : 42

I.A. Marks : 25
Exam Hours: 03
Exam Marks: 50

PART - A

Design, develop, and implement the following programs in C / C++

1. Program to recursively subdivide a tetrahedron to form 3D Sierpinski gasket. The number of recursive steps is to be specified by the user.

2. Program to implement Liang-Barsky line clipping algorithm.
3. Program to draw a color cube and spin it using OpenGL transformation matrices.
4. Program to create a house like figure and rotate it about a given fixed point using OpenGL functions.
5. Program to implement the Cohen-Sutherland line-clipping algorithm. Make provision to specify the input line, window for clipping and view port for displaying the clipped image.
6. Program to create a cylinder and a parallelepiped by extruding a circle and quadrilateral respectively. Allow the user to specify the circle and the quadrilateral.
7. Program, using OpenGL functions, to draw a simple shaded scene consisting of a tea pot on a table. Define suitably the position and properties of the light source along with the properties of the properties of the surfaces of the solid object used in the scene.
8. Program to draw a color cube and allow the user to move the camera suitably to experiment with perspective viewing. Use OpenGL functions.
9. Program to fill any given polygon using scan-line area filling algorithm. (Use appropriate data structures.)
10. Program to display a set of values {fij} as a rectangular mesh.

PART - B

Develop a suitable Graphics package to implement the skills learnt in the theory and the exercises indicated in Part A. Use the OpenGL.

Note:

1. Any question from Part A may be asked in the examination.
2. A report of about 10 – 12 pages on the package developed in Part B, duly certified by the department must be submitted during examination.

Instructions:

In the examination, one exercise from Part A is to be asked for a total of 30 marks. The package developed under Part B has to be evaluated for a total of 20 marks.

UNIX SYSTEM PROGRAMMING AND COMPILER DESIGN LABORATORY

Subject Code: 10CSL68
Hours/Week : 03
Total Hours : 42

I.A. Marks : 25
Exam Hours: 03
Exam Marks: 50

List of Experiments for USP: Design, develop, and execute the following programs

1. Write a C/C++ POSIX compliant program to check the following limits:
 - (i) No. of clock ticks
 - (ii) Max. no. of child processes
 - (iii) Max. path length
 - (iv) Max. no. of characters in a file name
 - (v) Max. no. of open files/ process
2. Write a C/C++ POSIX compliant program that prints the POSIX defined configuration options supported on any given system using feature test macros.
3. Consider the last 100 bytes as a region. Write a C/C++ program to check whether the region is locked or not. If the region is locked, print pid of the process which has locked. If the region is not locked, lock the region with an exclusive lock, read the last 50 bytes and unlock the region.
4. Write a C/C++ program which demonstrates interprocess communication between a reader process and a writer process. Use mkfifo, open, read, write and close APIs in your program.
5.
 - a) Write a C/C++ program that outputs the contents of its Environment list
 - b) Write a C / C++ program to emulate the unix **ln** command
6. Write a C/C++ program to illustrate the race condition.
7. Write a C/C++ program that creates a zombie and then calls system to execute the **ps** command to verify that the process is zombie.
8. Write a C/C++ program to avoid zombie process by forking twice.
9. Write a C/C++ program to implement the **system** function.

10. Write a C/C++ program to set up a real-time clock interval timer using the **alarm** API.

List of Experiments for Compiler Design: Design, develop, and execute the following programs.

11. Write a C program to implement the syntax-directed definition of “if E then S1” and “if E then S1 else S2”. (Refer Fig. 8.23 in the text book prescribed for 06CS62 Compiler Design, Alfred V Aho, Ravi Sethi, and Jeffrey D Ullman: Compilers- Principles, Techniques and Tools, 2nd Edition, Pearson Education, 2007).
12. Write a yacc program that accepts a regular expression as input and produce its parse tree as output.

Note: In the examination *each* student picks one question from the lot of *all* 12 questions.

VII SEMESTER

OBJECT-ORIENTED MODELING AND DESIGN

Subject Code: 10CS71
Hours/Week : 04
Total Hours : 52

I.A. Marks : 25
Exam Hours: 03
Exam Marks: 100

PART – A

UNIT – 1

7 Hours

Introduction, Modeling Concepts, class Modeling: What is Object Orientation? What is OO development? OO themes; Evidence for usefulness of OO development; OO modeling history

Modeling as Design Technique: Modeling; abstraction; The three models.

Class Modeling: Object and class concepts; Link and associations concepts; Generalization and inheritance; A sample class model; Navigation of class models; Practical tips.

UNIT – 2

6 Hours

Advanced Class Modeling, State Modeling: Advanced object and class concepts; Association ends; N-ary associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived data; Packages; Practical tips.

State Modeling: Events, States, Transitions and Conditions; State diagrams; State diagram behavior; Practical tips.

UNIT – 3

6 Hours

Advanced State Modeling, Interaction Modeling: Advanced State Modeling: Nested state diagrams; Nested states; Signal generalization; Concurrency; A sample state model; Relation of class and state models; Practical tips.

Interaction Modeling: Use case models; Sequence models; Activity models.

Use case relationships; Procedural sequence models; Special constructs for activity models.

UNIT – 4

7 Hours

Process Overview, System Conception, Domain Analysis: Process Overview: Development stages; Development life cycle.

System Conception: Devising a system concept; Elaborating a concept; Preparing a problem statement.

Domain Analysis: Overview of analysis; Domain class model; Domain state model; Domain interaction model; Iterating the analysis.

PART – B

UNIT – 5

7 Hours

Application Analysis, System Design: Application Analysis: Application interaction model; Application class model; Application state model; Adding operations.

Overview of system design; Estimating performance; Making a reuse plan; Breaking a system in to sub-systems; Identifying concurrency; Allocation of sub-systems; Management of data storage; Handling global resources; Choosing a software control strategy; Handling boundary conditions; Setting the trade-off priorities; Common architectural styles; Architecture of the ATM system as the example.

UNIT – 6

7 Hours

Class Design, Implementation Modeling, Legacy Systems: Class Design: Overview of class design; Bridging the gap; Realizing use cases; Designing algorithms; Recursing downwards, Refactoring; Design optimization; Reification of behavior; Adjustment of inheritance; Organizing a class design; ATM example.

Implementation Modeling: Overview of implementation; Fine-tuning classes; Fine-tuning generalizations; Realizing associations; Testing.

Legacy Systems: Reverse engineering; Building the class models; Building the interaction model; Building the state model; Reverse engineering tips; Wrapping; Maintenance.

UNIT – 7

6 Hours

Design Patterns – 1: What is a pattern and what makes a pattern? Pattern categories; Relationships between patterns; Pattern description
Communication Patterns: Forwarder-Receiver; Client-Dispatcher-Server; Publisher-Subscriber.

UNIT – 8

6 Hours

Design Patterns – 2, Idioms: Management Patterns: Command processor; View handler.

Idioms: Introduction; what can idioms provide? Idioms and style; Where to find idioms; Counted Pointer example

Text Books:

1. Michael Blaha, James Rumbaugh: Object-Oriented Modeling and Design with UML, 2nd Edition, Pearson Education, 2005.
(Chapters 1 to 17, 23)
2. Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal: Pattern-Oriented Software Architecture, A System of Patterns, Volume 1, John Wiley and Sons, 2007.
(Chapters 1, 3.5, 3.6, 4)

Reference Books:

1. Grady Booch et al: Object-Oriented Analysis and Design with Applications, 3rd Edition, Pearson Education, 2007.
2. Brahma Dathan, Sarnath Ramnath: Object-Oriented Analysis, Design, and Implementation, Universities Press, 2009.
3. Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado: UML 2 Toolkit, Wiley- Dreamtech India, 2004.
4. Simon Bennett, Steve McRobb and Ray Farmer: Object-Oriented Systems Analysis and Design Using UML, 2nd Edition, Tata McGraw-Hill, 2002.

EMBEDDED COMPUTING SYSTEMS

Sub Code: 10CS72
Hrs/Week: 04
Total Hrs: 52

IA Marks :25
Exam Hours :03
Exam Marks :100

PART- A

UNIT – 1 **6 Hours**

Embedded Computing: Introduction, Complex Systems and Microprocessors, Embedded Systems Design Process, Formalism for System design

Design Example: Model Train Controller.

UNIT – 2 **7 Hours**

Instruction Sets, CPUs: Preliminaries, ARM Processor, Programming Input and Output, Supervisor mode, Exceptions, Traps, Coprocessors, Memory Systems Mechanisms, CPU Performance, CPU Power Consumption. Design Example: Data Compressor.

UNIT – 3 **6 Hours**

Bus-Based Computer Systems: CPU Bus, Memory Devices, I/O devices, Component Interfacing, Designing with Microprocessor, Development and Debugging, System-Level Performance Analysis

Design Example: Alarm Clock.

UNIT – 4 **7 Hours**

Program Design and Analysis: Components for embedded programs, Models of programs, Assembly, Linking and Loading, Basic Compilation Techniques, Program optimization, Program-Level performance analysis, Software performance optimization, Program-Level energy and power analysis, Analysis and optimization of program size, Program validation and testing. Design Example: Software modem.

PART- B

UNIT – 5 **6 Hours**

Real Time Operating System (RTOS) Based Design – 1: Basics of OS, Kernel, types of OSs, tasks, processes, Threads, Multitasking and Multiprocessing, Context switching, Scheduling Policies, Task Communication, Task Synchronization.

UNIT – 6 **6 Hours**

RTOS-Based Design - 2: Inter process Communication mechanisms, Evaluating OS performance, Choice of RTOS, Power Optimization. Design Example: Telephone Answering machine

UNIT – 7

7 Hours

Distributed Embedded Systems: Distributed Network Architectures, Networks for Embedded Systems: I2C Bus, CAN Bus, SHARC Link Ports, Ethernet, Myrinet, Internet, Network Based Design. Design Example: Elevator Controller.

UNIT – 8

7 Hours

Embedded Systems Development Environment: The Integrated Development Environment, Types of File generated on Cross Compilation, Dis-assembler /Decompiler, Simulators, Emulators, and Debugging, Target Hardware Debugging.

Text Books:

1. Wayne Wolf: Computers as Components, Principles of Embedded Computing Systems Design, 2nd Edition, Elsevier, 2008.
2. Shibu K V: Introduction to Embedded Systems, Tata McGraw Hill, 2009
(Chapters 10, 13)

Reference Books:

1. James K. Peckol: Embedded Systems, A contemporary Design Tool, Wiley India, 2008
2. Tammy Neorgaard: Embedded Systems Architecture, Elsevier, 2005.

PROGRAMMING THE WEB

Subject Code: 10CS73

I.A. Marks : 25

Hours/Week : 04

Exam Hours: 03

Total Hours : 52

Exam Marks: 100

UNIT – 1

6 Hours

Fundamentals of Web, XHTML – 1: Internet, WWW, Web Browsers and Web Servers, URLs, MIME, HTTP, Security, The Web Programmers Toolbox.

XHTML: Basic syntax, Standard structure, Basic text markup, Images, Hypertext Links.

UNIT – 2**7 Hours**

XHTML – 2, CSS: XHTML (continued): Lists, Tables, Forms, Frames
CSS: Introduction, Levels of style sheets, Style specification formats, Selector forms, Property value forms, Font properties, List properties, Color, Alignment of text, The box model, Background images, The and <div> tags, Conflict resolution.

UNIT – 3**6 Hours**

Javascript: Overview of Javascript, Object orientation and Javascript, Syntactic characteristics, Primitives, operations, and expressions, Screen output and keyboard input, Control statements, Object creation and modification, Arrays, Functions, Constructors, Pattern matching using regular expressions, Errors in scripts, Examples.

UNIT – 4**7 Hours**

Javascript and HTML Documents, Dynamic Documents with Javascript: The Javascript execution environment, The Document Object Model, Element access in Javascript, Events and event handling, Handling events from the Body elements, Button elements, Text box and Password elements, The DOM 2 event model, The navigator object, DOM tree traversal and modification. Introduction to dynamic documents, Positioning elements, Moving elements, Element visibility, Changing colors and fonts, Dynamic content, Stacking elements, Locating the mouse cursor, Reacting to a mouse click, Slow movement of elements, Dragging and dropping elements.

PART - B**UNIT – 5****6 Hours**

XML: Introduction, Syntax, Document structure, Document type definitions, Namespaces, XML schemas, Displaying raw XML documents, Displaying XML documents with CSS, XSLT style sheets, XML processors, Web services.

UNIT – 6**7 Hours**

Perl, CGI Programming: Origins and uses of Perl, Scalars and their operations, Assignment statements and simple input and output, Control statements, Fundamentals of arrays, Hashes, References, Functions, Pattern matching, File input and output; Examples.
The Common Gateway Interface; CGI linkage; Query string format; CGI.pm module; A survey example; Cookies.
Database access with Perl and MySQL

UNIT – 7**6 Hours**

PHP: Origins and uses of PHP, Overview of PHP, General syntactic characteristics, Primitives, operations and expressions, Output, Control

statements, Arrays, Functions, Pattern matching, Form handling, Files, Cookies, Session tracking, Database access with PHP and MySQL.

UNIT – 8

7 Hours

Ruby, Rails: Origins and uses of Ruby, Scalar types and their operations, Simple input and output, Control statements, Arrays, Hashes, Methods, Classes, Code blocks and iterators, Pattern matching.

Overview of Rails, Document requests, Processing forms, Rails applications with Databases, Layouts.

Text Books:

1. Robert W. Sebesta: Programming the World Wide Web, 4th Edition, Pearson Education, 2008.
(Listed topics only from Chapters 1 to 9, 11 to 15)

Reference Books:

1. M. Deitel, P.J. Deitel, A. B. Goldberg: Internet & World Wide Web How to Program, 4th Edition, Pearson Education, 2004.
2. Chris Bates: Web Programming Building Internet Applications, 3rd Edition, Wiley India, 2007.
3. Xue Bai et al: The web Warrior Guide to Web Programming, Cengage Learning, 2003.

JAVA AND J2EE

Subject Code:10CS753

Hours/Week: 4

Total Hours: 52

IA Marks: 25

Exam Marks: 100

Exam Hours: 3

PART - A

UNIT – 1

6 Hours

Introduction to Java: Java and Java applications; Java Development Kit (JDK); Java is interpreted, Byte Code, JVM; Object-oriented programming; Simple Java programs.

Data types and other tokens: Boolean variables, int, long, char, operators, arrays, white spaces, literals, assigning values; Creating and destroying objects; Access specifiers.

Operators and Expressions: Arithmetic Operators, Bitwise operators, Relational operators, The Assignment Operator, The ? Operator; Operator Precedence; Logical expression; Type casting; Strings

Control Statements: Selection statements, iteration statements, Jump Statements.

UNIT – 2

6 Hours

Classes, Inheritance, Exceptions, Applets : Classes: Classes in Java; Declaring a class; Class name; Super classes; Constructors; Creating instances of class; Inner classes.

Inheritance: Simple, multiple, and multilevel inheritance; Overriding, overloading.

Exception handling: Exception handling in Java.

The Applet Class: Two types of Applets; Applet basics; Applet Architecture; An Applet skeleton; Simple Applet display methods; Requesting repainting; Using the Status Window; The HTML APPLLET tag; Passing parameters to Applets; getDocumentbase() and getCodebase(); ApletContext and showDocument(); The AudioClip Interface; The AppletStub Interface; Output to the Console.

UNIT – 3

7 Hours

Multi Threaded Programming, Event Handling: Multi Threaded Programming: What are threads? How to make the classes threadable; Extending threads; Implementing runnable; Synchronization; Changing state of the thread; Bounded buffer problems, read-write problem, producer-consumer problems.

Event Handling: Two event handling mechanisms; The delegation event model; Event classes; Sources of events; Event listener interfaces; Using the delegation event model; Adapter classes; Inner classes.

UNIT – 4

7 Hours

Swings: Swings: The origins of Swing; Two key Swing features; Components and Containers; The Swing Packages; A simple Swing Application; Create a Swing Applet; JLabel and ImageIcon; JTextField;The Swing Buttons; JTabbedPane; JScrollPane; JList; JComboBox; JTable.

PART – B

UNIT – 5

6 Hours

Java 2 Enterprise Edition Overview, Database Access: Overview of J2EE and J2SE

The Concept of JDBC; JDBC Driver Types; JDBC Packages; A Brief Overview of the JDBC process; Database Connection; Associating the JDBC/ODBC Bridge with the Database; Statement Objects; ResultSet; Transaction Processing; Metadata, Data types; Exceptions.

UNIT – 6

7 Hours

Servlets: Background; The Life Cycle of a Servlet; Using Tomcat for Servlet Development; A simple Servlet; The Servlet API; The javax.servlet Package; Reading Servlet Parameter; The javax.servlet.http package; Handling HTTP Requests and Responses; Using Cookies; Session Tracking.

UNIT – 7

6 Hours

JSP, RMI: Java Server Pages (JSP): JSP, JSP Tags, Tomcat, Request String, User Sessions, Cookies, Session Objects.

Java Remote Method Invocation: Remote Method Invocation concept; Server side, Client side.

UNIT – 8

7 Hours

Enterprise Java Beans: Enterprise java Beans; Deployment Descriptors; Session Java Bean, Entity Java Bean; Message-Driven Bean; The JAR File.

Text Books:

1. Herbert Schildt: Java The Complete Reference, 7th Edition, Tata McGraw Hill, 2007.
(Chapters 1, 2, 3, 4, 5, 6, 8, 10, 11, 21, 22, 29, 30, 31)
2. Jim Keogh: J2EE - The Complete Reference, Tata McGraw Hill, 2007.
(Chapters 5, 6, 11, 12, 15)

Reference Books:

1. Y. Daniel Liang: Introduction to JAVA Programming, 7th Edition, Pearson Education, 2007.
2. Stephanie Bodoff et al: The J2EE Tutorial, 2nd Edition, Pearson Education, 2004.

DATA WAREHOUSING AND DATA MINING

Subject Code: 10CS755

Hours/Week : 04

Total Hours : 52

I.A. Marks : 25

Exam Hours: 03

Exam Marks: 100

PART – A

UNIT – 1

6 Hours

Data Warehousing:

Introduction, Operational Data Stores (ODS), Extraction Transformation Loading (ETL), Data Warehouses. Design Issues, Guidelines for Data Warehouse Implementation, Data Warehouse Metadata

UNIT – 2

6 Hours

Online Analytical Processing (OLAP): Introduction, Characteristics of OLAP systems, Multidimensional view and Data cube, Data Cube Implementations, Data Cube operations, Implementation of OLAP and overview on OLAP Softwares.

UNIT – 3

6 Hours

Data Mining: Introduction, Challenges, Data Mining Tasks, Types of Data, Data Preprocessing, Measures of Similarity and Dissimilarity, Data Mining Applications

UNIT – 4

8 Hours

Association Analysis: Basic Concepts and Algorithms: Frequent Itemset Generation, Rule Generation, Compact Representation of Frequent Itemsets, Alternative methods for generating Frequent Itemsets, FP Growth Algorithm, Evaluation of Association Patterns

PART - B

UNIT – 5

6 Hours

Classification -1 : Basics, General approach to solve classification problem, Decision Trees, Rule Based Classifiers, Nearest Neighbor Classifiers.

UNIT – 6

6 Hours

Classification - 2: Bayesian Classifiers, Estimating Predictive accuracy of classification methods, Improving accuracy of classification methods, Evaluation criteria for classification methods, Multiclass Problem.

UNIT – 7**8 Hours**

Clustering Techniques: Overview, Features of cluster analysis, Types of Data and Computing Distance, Types of Cluster Analysis Methods, Partitional Methods, Hierarchical Methods, Density Based Methods, Quality and Validity of Cluster Analysis

UNIT – 8**6 Hours**

Web Mining: Introduction, Web content mining, Text Mining, Unstructured Text, Text clustering, Mining Spatial and Temporal Databases.

Text Books:

1. Pang-Ning Tan, Michael Steinbach, Vipin Kumar: Introduction to Data Mining, Pearson Education, 2005.
2. G. K. Gupta: Introduction to Data Mining with Case Studies, 3rd Edition, PHI, New Delhi, 2009.

Reference Books:

1. Arun K Pujari: Data Mining Techniques 2nd Edition, Universities Press, 2009.
2. Jiawei Han and Micheline Kamber: Data Mining - Concepts and Techniques, 2nd Edition, Morgan Kaufmann Publisher, 2006.
3. Alex Berson and Stephen J. Smith: Data Warehousing, Data Mining, and OLAP Computing, Mc GrawHill Publisher, 1997.

C# PROGRAMMING AND .NET

Subject Code: 10CS761

Hours/Week : 04

Total Hours : 52

I.A. Marks : 25

Exam Hours: 03

Exam Marks: 100

PART – A

UNIT – 1

6 Hours

The philosophy of .NET: Understanding the Previous State of Affairs, The .NET Solution, The Building Block of the .NET Platform (CLR,CTS, and CLS), The Role of the .NET Base Class Libraries, What C# Brings to the Table, An Overview of .NET Binaries (aka Assemblies), the Role of the Common Intermediate Language , The Role of .NET Type Metadata, The Role of the Assembly Manifest, Compiling CIL to Platform –Specific Instructions, Understanding the Common Type System, Intrinsic CTS Data Types, Understanding the Common Language Specification, Understanding the Common Language Runtime A tour of the .NET Namespaces, Increasing Your Namespace Nomenclature, Deploying the .NET Runtime

UNIT – 2

6 Hours

Building C# Applications: The Role of the Command Line Compiler (csc.exe), Building C # Application using csc.exe Working with csc.exe Response Files, Generating Bug Reports , Remaining C# Compiler Options, The Command Line Debugger (cordbg.exe) Using the, Visual Studio .NET

IDE, Other Key Aspects of the VS.NET IDE, C# “Preprocessor:” Directives, An Interesting Aside: The System. Environment Class

UNIT – 3

8 Hours

C# Language Fundamentals: The Anatomy of a Basic C# Class, Creating objects: Constructor Basics, The Composition of a C# Application, Default Assignment and Variable Scope, The C# Member Initialization Syntax, Basic Input and Output with the Console Class, Understanding Value Types and Reference Types, The Master Node: System, Object, The System Data Types (and C# Aliases), Converting Between Value Types and Reference Types: Boxing and Unboxing, Defining Program Constants, C# Iteration Constructs, C# Controls Flow Constructs, The Complete Set of C# Operators, Defining Custom Class Methods, Understating Static Methods, Methods Parameter Modifies, Array Manipulation in C #, String Manipulation in C#, C# Enumerations, Defining Structures in C#, Defining Custom Namespaces

UNIT – 4

6 Hours

Object- Oriented Programming with C#: Forms Defining of the C# Class, Definition the “Default Public Interface” of a Type, Recapping the Pillars of OOP, The First Pillars: C#’s Encapsulation Services, Pseudo- Encapsulation: Creating Read-Only Fields, The Second Pillar: C#’s Inheritance Supports, keeping Family Secrets: The “Protected” Keyword, Nested Type Definitions, The Third Pillar: C #’s Polymorphic Support, Casting Between.

PART – B

UNIT – 5

6 Hours

Exceptions and Object Lifetime: Ode to Errors, Bugs, and Exceptions, The Role of .NET Exception Handling, the System. Exception Base Class, Throwing a Generic Exception, Catching Exception, CLR System – Level Exception(System. System Exception), Custom Application-Level Exception(System. System Exception), Handling Multiple Exception, The Family Block, the Last Chance Exception Dynamically Identifying Application – and System Level Exception Debugging System Exception Using VS. NET, Understanding Object Lifetime, the CIT of “new”, The Basics of Garbage Collection,, Finalization a Type, The Finalization Process, Building an Ad Hoc Destruction Method, Garbage Collection Optimizations, The System. GC Type.

UNIT – 6

6 Hours

Interfaces and Collections: Defining Interfaces Using C# Invoking Interface Members at the object Level, Exercising the Shapes Hierarchy, Understanding Explicit Interface Implementation, Interfaces As Polymorphic Agents, Building Interface Hierarchies, Implementing, Implementation, Interfaces Using VS .NET, understanding the IConvertible Interface, Building a Custom Enumerator (IEnumerable and Enumerator), Building Cloneable objects (ICloneable), Building Comparable Objects (I Comparable), Exploring the system. Collections Namespace, Building a Custom Container (Retrofitting the Cars Type)

UNIT – 7

8 Hours

Callback Interfaces, Delegates, and Events, Advanced Techniques: Understanding Callback Interfaces, Understanding the .NET Delegate Type, Members of System. Multicast Delegate, The Simplest Possible Delegate Example, Building More a Elaborate Delegate Example, Understanding Asynchronous Delegates, Understanding (and Using)Events. The Advances Keywords of C#, A Catalog of C# Keywords Building a Custom Indexer, A Variation of the Cars Indexer Internal Representation of Type Indexer . Using C# Indexer from VB .NET. Overloading operators, The Internal Representation of Overloading Operators, interacting with Overload Operator from Overloaded- Operator- Challenged Languages, Creating Custom Conversion Routines, Defining Implicit Conversion Routines, The Internal Representations of Customs Conversion Routines

UNIT – 8

6 Hours

Understanding .NET Assemblies: Problems with Classic COM Binaries, An Overview of .NET Assembly, Building a Simple File Test Assembly, A C#. Client Application, A Visual Basic .NET Client Application, Cross Language Inheritance, Exploring the CarLibrary's, Manifest, Exploring the CarLibrary's Types, Building the Multifile Assembly, Using Assembly, Understanding Private Assemblies, Probing for Private Assemblies (The Basics), Private A Assemblies XML Configurations Files, Probing for Private Assemblies (The Details), Understanding Shared Assembly, Understanding Shared Names, Building a Shared Assembly, Understanding Delay Signing, Installing/Removing Shared Assembly, Using a Shared Assembly

Text Books:

1. Andrew Troelsen: Pro C# with .NET 3.0, 4th Edition, Wiley India, 2009.
Chapters: 1 to 11 (up to pp.369)
2. E. Balagurusamy: Programming in C#, 2nd Edition, Tata McGraw Hill, 2008.

(Programming Examples 3.7, 3.10, 5.5, 6.1, 7.2, 7.4, 7.5, 7.6, 8.1, 8.2, 8.3, 8.5, 8.7, 8.8, 9.1, 9.2, 9.3, 9.4, 10.2, 10.4, 11.2, 11.4, 12.1, 12.4, 12.5, 12.6, 13.1, 13.2, 13.3, 13.6, 14.1, 14.2, 14.4, 15.2, 15.3, 16.1, 16.2, 16.3, 18.3, 18.5.18.6)

Reference Books:

1. Tom Archer: Inside C#, WP Publishers, 2001.
2. Herbert Schildt: C# The Complete Reference, Tata McGraw Hill, 2004.

DIGITAL IMAGE PROCESSING

Subject Code: 10CS762

Hours/Week : 04

Total Hours : 52

I.A. Marks : 25

Exam Hours: 03

Exam Marks: 100

PART – A

UNIT – 1

6 Hours

Digitized Image and its properties: Basic concepts, Image digitization, Digital image properties

UNIT – 2

7 Hours

Image Preprocessing: Image pre-processing: Brightness and geometric transformations, local preprocessing.

UNIT – 3

7 Hours

Segmentation – 1: Thresholding, Edge-based segmentation.

UNIT – 4

7 Hours

Segmentation – 2: Region based segmentation, Matching.

PART – B

UNIT – 5

7 Hours

Image Enhancement: Image enhancement in the spatial domain: Background, Some basic gray level transformations, Histogram processing, Enhancement using arithmetic/ logic operations, Basics of spatial filtering, Smoothing spatial filters, Sharpening spatial filters. Image enhancement in the frequency domain: Background, Introduction to the Fourier transform and the frequency domain, Smoothing Frequency-Domain filters, Sharpening Frequency Domain filters, Homomorphic filtering.

UNIT – 6

6 Hours

Image Compression: Image compression: Fundamentals, Image compression models, Elements of information theory, Error-Free Compression, Lossy compression.

UNIT – 7**7 Hours**

Shape representation: Region identification, Contour-based shape representation and description, Region based shape representation and description, Shape classes.

UNIT – 8**6 Hours**

Morphology: Basic morphological concepts, Morphology principles, Binary dilation and erosion, Gray-scale dilation and erosion, Morphological segmentation and watersheds

Text Books:

1. Milan Sonka, Vaclav Hlavac and Roger Boyle: Image Processing, Analysis and Machine Vision, 2nd Edition, Thomson Learning, 2001.
(Chapters 2, 4.1 to 4.3, 5.1 to 5.4, 6, 11.1 to 11.4, 11.7)
2. Rafael C Gonzalez and Richard E Woods: Digital Image Processing, 3rd Edition, Pearson Education, 2003.
(Chapters 3.1 to 3.7, 4.1 to 4.5, 8.1 to 8.5)

Reference Books:

1. Anil K Jain, “Fundamentals of Digital Image Processing”, PHI, 1997, Indian Reprint 2009.
2. B.Chanda, D Dutta Majumder, “Digital Image Processing and Analysis”, PHI, 2002.

STORAGE AREA NETWORKS

Subject Code: 10CS765

Hours/Week : 04

Total Hours : 52

I.A. Marks : 25

Exam Hours: 03

Exam Marks: 100

PART –A

UNIT - 1

7 Hours

Introduction to Information Storage and Management, Storage System Environment: Information Storage, Evolution of Storage Technology and Architecture, Data Center Infrastructure, Key Challenges in Managing Information, Information Lifecycle

Components of Storage System Environment, Disk Drive Components, Disk Drive Performance, Fundamental Laws Governing Disk Performance, Logical Components of the Host, Application Requirements and Disk Performance.

UNIT - 2

6 Hours

Data Protection, Intelligent Storage system: Implementation of RAID, RAID Array Components, RAID Levels, RAID Comparison, RAID Impact on Disk Performance, Hot Spares

Components of an Intelligent Storage System, Intelligent Storage Array

UNIT - 3

7 Hours

Direct-Attached Storage, SCSI, and Storage Area Networks: Types of DAS, DAS Benefits and Limitations, Disk Drive Interfaces, Introduction to Parallel SCSI, Overview of Fibre Channel, The SAN and Its Evolution, Components of SAN, FC Connectivity, Fibre Channel Ports, Fibre Channel Architecture, Zoning, Fibre Channel Login Types, FC Topologies.

UNIT - 4

6 Hours

NAS, IP SAN: General – Purpose Service vs. NAS Devices, Benefits of NAS, NAS File I / O, Components of NAS, NAS Implementations, NAS

File-Sharing Protocols, NAS I/O Operations, Factors Affecting NAS Performance and Availability. iSCSI, FCIP.

PART - B

UNIT - 5

6 Hours

Content-Addressed Storage, Storage Virtualization: Fixed Content and Archives, Types of Archive, Features and Benefits of CAS, CAS Architecture, Object Storage and Retrieval in CAS, CAS Examples
Forms of Virtualization, SNIA Storage Virtualization Taxonomy, Storage Virtualizations Configurations, Storage Virtualization Challenges, Types of Storage Virtualization

UNIT - 6

6 Hours

Business Continuity, Backup and Recovery: Information Availability, BC Terminology, BC Planning Lifecycle, Failure Analysis, Business Impact Analysis, BC Technology Solutions.
Backup Purpose, Backup Considerations, Backup Granularity, Recovery Considerations, Backup Methods, Backup Process, Backup and restore Operations, Backup Topologies, Backup in NAS Environments, Backup Technologies.

UNIT - 7

7 Hours

Local Replication, Remote Replication: Source and Target, Uses of Local Replicas, Data Consistency, Local Replication Technologies, Restore and Restart Considerations, Creating Multiple Replicas, Management Interface, Modes of Remote Replication, Remote Replication Technologies, Network Infrastructure.

UNIT - 8

7 Hours

Securing the Storage Infrastructure, Managing the Storage Infrastructure: Storage Security Framework, Risk Triad, Storage Security Domains, Security Implementations in Storage Networking
Monitoring the Storage Infrastructure, Storage Management Activities, Storage Infrastructure Management Challenges, Developing an Ideal Solution.

Text Books:

1. G. Somasundaram, Alok Shrivastava (Editors): Information Storage and Management, EMC Education Services, Wiley India, 2009.

Reference Books:

1. Ulf Troppens, Rainer Erkens and Wolfgang Muller: Storage Networks Explained, Wiley India, 2003.
2. Rebert Spalding: Storage Networks, The Complete Reference, Tata McGraw Hill, 2003.

Networks Laboratory

Subject Code: 10CSL77

Hours/Week : 03

Total Hours : 42

I.A. Marks : 25

Exam Hours: 03

Exam Marks: 50

Note: Student is required to solve one problem from PART-A and one problem from PART-B. The questions are allotted based on lots. Both questions carry equal marks.

PART A – Simulation Exercises

The following experiments shall be conducted using either NS228/OPNET or any other suitable simulator.

1. Simulate a three nodes point – to – point network with duplex links between them. Set the queue size and vary the bandwidth and find the number of packets dropped.
2. Simulate a four node point-to-point network with the links connected as follows:
n0 – n2, n1 – n2 and n2 – n3. Apply TCP agent between n0-n3 and UDP between n1-n3. Apply relevant applications over TCP and UDP agents changing the parameter and determine the number of packets sent by TCP / UDP.
3. Simulate the transmission of ping messages over a network topology consisting of 6 nodes and find the number of packets dropped due to congestion.
4. Simulate an Ethernet LAN using n nodes (6-10), change error rate and data rate and compare throughput.
5. Simulate an Ethernet LAN using n nodes and set multiple traffic nodes and plot congestion window for different source / destination.
6. Simulate simple ESS and with transmitting nodes in wire-less LAN by simulation and determine the performance with respect to transmission of packets.

PART-B

Implement the following in C/C++:

7. Write a program for error detecting code using CRC-CCITT (16- bits).
8. Write a program for distance vector algorithm to find suitable path for transmission.
9. Using TCP/IP sockets, write a client – server program to make the client send the file name and to make the server send back the contents of the requested file if present.

10. Implement the above program using as message queues or FIFOs as IPC channels.
11. Write a program for simple RSA algorithm to encrypt and decrypt the data.
12. Write a program for congestion control using leaky bucket algorithm.

Note:

In the examination, a combination of one problem has to be asked from Part A for a total of 25 marks and one problem from Part B has to be asked for a total of 25 marks. The choice must be based on random selection from the entire lots.

Web Programming Laboratory

Subject Code: 10CSL78

Hours/Week : 03

Total Hours : 42

I.A. Marks : 25

Exam Hours: 03

Exam Marks: 50

1. Develop and demonstrate a XHTML file that includes Javascript script for the following problems:
 - a) Input: A number n obtained using prompt
Output: The first n Fibonacci numbers
 - b) Input: A number n obtained using prompt
Output: A table of numbers from 1 to n and their squares using **alert**
2. a) Develop and demonstrate, using Javascript script, a XHTML document that collects the USN (the valid format is: A digit from 1 to 4 followed by two upper-case characters followed by two digits followed by two upper-case characters followed by three digits; no embedded spaces allowed) of the user. Event handler must be included for the form element that collects this information to validate the input. Messages in the alert windows must be produced when errors are detected.
 - b) Modify the above program to get the current semester also (restricted to be a number from 1 to 8)
3. a) Develop and demonstrate, using Javascript script, a XHTML document that contains three short paragraphs of text, stacked on top of each other, with only enough of each showing so that the mouse cursor can be placed over some part of them. When the cursor is placed over the exposed part of any paragraph, it should rise to the top to become completely visible.
 - b) Modify the above document so that when a paragraph is moved from the top stacking position, it returns to its original position rather than to the bottom.
4. a) Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include

USN, Name, Name of the College, Branch, Year of Joining, and e-mail id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.

- b) Create an XSLT style sheet for one student element of the above document and use it to create a display of that element.
5. a) Write a Perl program to display various Server Information like Server Name, Server Software, Server protocol, CGI Revision etc.
b) Write a Perl program to accept UNIX command from a HTML form and to display the output of the command executed.
6. a) Write a Perl program to accept the User Name and display a greeting message randomly chosen from a list of 4 greeting messages.
b) Write a Perl program to keep track of the number of visitors visiting the web page and to display this count of visitors, with proper headings.
7. Write a Perl program to display a digital clock which displays the current time of the server.
8. Write a Perl program to insert name and age information entered by the user into a table created using MySQL and to display the current contents of this table.
9. Write a PHP program to store current date-time in a COOKIE and display the 'Last visited on' date-time on the web page upon reopening of the same page.
10. Write a PHP program to store page views count in SESSION, to increment the count on each refresh, and to show the count on web page.
11. Create a XHTML form with Name, Address Line 1, Address Line 2, and E-mail text fields. On submitting, store the values in MySQL table. Retrieve and display the data based on Name.
12. Build a Rails application to accept book information viz. Accession number, title, authors, edition and publisher from a web page and store the information in a database and to search for a book with the title specified by the user and to display the search results with proper headings.

Note: In the examination *each* student picks one question from the lot of *all* 12 questions.

SYSTEM MODELING AND SIMULATION

Sub Code: 10CS82
Hrs/Week: 04
Total Hrs: 52

IA Marks : 25
Exam Hours : 03
Exam Marks : 100

PART – A

UNIT – 1 8 Hours

Introduction: When simulation is the appropriate tool and when it is not appropriate; Advantages and disadvantages of Simulation; Areas of application; Systems and system environment; Components of a system; Discrete and continuous systems; Model of a system; Types of Models; Discrete-Event System Simulation; Steps in a Simulation Study. The basics of Spreadsheet simulation, Simulation example: Simulation of queuing systems in a spreadsheet.

UNIT – 2 6 Hours

General Principles, Simulation Software: Concepts in Discrete-Event Simulation: The Event-Scheduling / Time-Advance Algorithm, World Views, Manual simulation Using Event Scheduling; List processing. Simulation in Java; Simulation in GPSS

UNIT – 3 6 Hours

Statistical Models in Simulation: Review of terminology and concepts; Useful statistical models; Discrete distributions; Continuous distributions; Poisson process; Empirical distributions.

UNIT – 4 6 Hours

Queuing Models: Characteristics of queuing systems; Queuing notation; Long-run measures of performance of queuing systems; Steady-state behavior of M/G/1 queue; Networks of queues; Rough-cut modeling: An illustration..

PART – B

UNIT – 5 8 Hours

Random-Number Generation, Random-Variate Generation: Properties of random numbers; Generation of pseudo-random numbers; Techniques for generating random numbers; Tests for Random Numbers Random-Variate Generation: Inverse transform technique; Acceptance-Rejection technique; Special properties.

UNIT – 6**6 Hours**

Input Modeling : Data Collection; Identifying the distribution with data; Parameter estimation; Goodness of Fit Tests; Fitting a non-stationary Poisson process; Selecting input models without data; Multivariate and Time-Series input models.

UNIT – 7**6 Hours**

Estimation of Absolute Performance: Types of simulations with respect to output analysis; Stochastic nature of output data; Absolute measures of performance and their estimation; Output analysis for terminating simulations; Output analysis for steady-state simulations.

UNIT – 8**6 Hours**

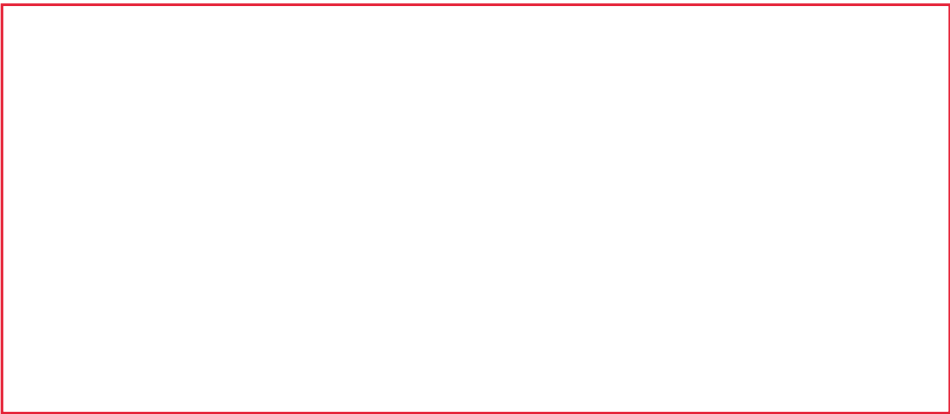
Verification, Calibration, and Validation; Optimization: Model building, verification and validation; Verification of simulation models; Calibration and validation of models, Optimization via Simulation

Text Books:

1. Jerry Banks, John S. Carson II, Barry L. Nelson, David M. Nicol: Discrete-Event System Simulation, 5th Edition, Pearson Education, 2010.
(Listed topics only from Chapters 1 to 12)

Reference Books:

1. Lawrence M. Leemis, Stephen K. Park: Discrete – Event Simulation: A First Course, Pearson Education, 2006.
2. Averill M. Law: Simulation Modeling and Analysis, 4th Edition, Tata McGraw-Hill, 2007.



WEB 2.0 AND RICH INTERNET APPLICATIONS

Sub Code: 10CS832

Hrs/ Week: 04

Total Hours: 52

IA Marks : 25

Exam Hours : 03

Exam Marks : 100

PART - A

UNIT – 1

6 Hours

Introduction, Ajax – 1: Web 2.0 and Rich Internet Applications, Overview of Ajax, Examples of usage of Ajax: Updating web page text, Chatting in real time, Dragging and dropping, Downloading images. Creating Ajax Applications: An example, Analysis of example ajax.html, Creating the JavaScript, Creating and opening the XMLHttpRequest object, Data download, Displaying the fetched data, Connecting to the server, Adding Server-side programming, Sending data to the server using GET and POST, Using Ajax together with XML.

UNIT – 2

7 Hours

Ajax – 2: Handling multiple XMLHttpRequest objects in the same page, Using two XMLHttpRequest objects, Using an array of XMLHttpRequest objects, Using inner functions, Downloading JavaScript, connecting to Google Suggest, Creating google.php, Downloading from other domains with Ajax, HTML header request and Ajax, Defeating caching, Examples. Building XML and working with XML in JavaScript, Getting the document element, Accessing any XML element, Handling whitespace in Firefox, Handling cross-browser whitespace, Accessing XML data directly, Validating XML, Further examples of Rich Internet Applications with Ajax.

UNIT – 3**6 Hours**

Ajax – 3: Drawing user’s attention to downloaded text, Styling text, colors and background using CSS, Setting element location in the web pages, Setting the stacking order of web page elements, Further examples of using Ajax. Displaying all the data in an HTML form, Working with PHP server variables, Getting the data in to array format, Wrapping applications in to a single PHP page, Validating input from the user, Validating integers and text, DOM, Appending new elements to a web page using the DOM and Ajax, Replacing elements using the DOM, Handling timeouts in Ajax, Downloading images with Ajax, Example programs.

UNIT – 4**7 Hours**

Flex – 1 : Introduction: Understanding Flex Application Technologies, Using Flex Elements, Working with Data Services (Loading Data at Runtime), The Differences between Traditional and Flex Web Applications, Understanding How Flex Applications Work, Understanding Flex and Flash Authoring. Building Applications with the Flex Framework: Using Flex Tool Sets, Creating Projects, Building Applications, Deploying Applications Framework Fundamentals: Understanding How Flex Applications Are Structured, Loading and Initializing Flex Applications, Understanding the Component Life Cycles, Loading One Flex Application into Another Flex Application, Differentiating Between Flash Player and the Flex Framework, Caching the Framework, Understanding Application Domains, Localization, Managing Layout: Flex Layout Overview, Making Fluid Interfaces, Putting It All Together.

PART B**UNIT – 5****7 Hours**

Flex – 2: MXML: Understanding MXML Syntax and Structure, Making MXML Interactive Working with UI Components: Understanding UI Components, Buttons, Value Selectors, Text Components, List-Based Controls, Pop-Up Controls, Navigators, Control Bars Customizing Application Appearance: Using Styles, Skinning components, Customizing the preloader, Themes, Runtime CSS

UNIT – 6**6 Hours**

Flex – 3: ActionScript: Using ActionScript, MXML and ActionScript Correlations, Understanding ActionScript Syntax, Variables and Properties, Inheritance, Interfaces, Handling Events, Error Handling, Using XML

UNIT – 7**7 Hours**

Flex – 4: Managing State: Creating States, Applying States, Defining States, Adding and Removing Components, Setting Properties, Setting Styles,

Setting Event Handlers, Using Action Scripts to Define States, Managing Object Creation Policies, Handling State Events, Understanding State Life Cycles, When To Use States.Using Effects and Transitions: Using Effects, Creating Custom Effects, Using Transitions, Creating Custom Transitions.

UNIT – 8

6 Hours

Flex – 5: Working with Data: Using Data Models, Data Binding, Enabling Data Binding for Custom Classes, Data Binding Examples, Building data binding proxies. Validating and Formatting Data: Validating user input, Formatting Data.

Text Books:

1. Steven Holzner: Ajax: A Beginner's Guide, Tata McGraw Hill, 2009.
(Listed topics from Chapters 3, 4, 6, 7, 11, 12)
2. Chafic Kazon and Joey Lott: Programming Flex 3, O'Reilly, June 2009.
(Listed topics from Chapters 1 to 8, 12 to 15)

Reference Books:

1. Jack Herrington and Emily Kim: Getting Started with Flex 3, O'Reilly, 1st Edition, 2008.
2. Michele E. Davis and John A. Phillips: Flex 3 - A Beginner's Guide, Tata McGraw-Hill, 2008.
3. Colin Moock: Essential Actionscript 3.0, O'Reilly Publications, 2007.
4. Nicholas C Zakas et al : Professional Ajax, 2nd Edition, Wrox/Wiley India, 2008.



INFORMATION AND NETWORK SECURITY

Subject Code: 10CS835

Hours/Week : 04

Total Hours : 52

I.A. Marks : 25

Exam Hours: 03

Exam Marks: 100

PART – A

UNIT 1

6 Hours

Planning for Security: Introduction; Information Security Policy, Standards, and Practices; The Information Security Blue Print; Contingency plan and a model for contingency plan

UNIT 2

6 Hours

Security Technology-1: Introduction; Physical design; Firewalls; Protecting Remote Connections

UNIT 3

6 Hours

Security Technology – 2: Introduction; Intrusion Detection Systems (IDS); Honey Pots, Honey Nets, and Padded cell systems; Scanning and Analysis Tools

UNIT 4

8 Hours

Cryptography: Introduction; A short History of Cryptography; Principles of Cryptography; Cryptography Tools; Attacks on Cryptosystems.

PART - B

UNIT 5**8 Hours**

Introduction to Network Security, Authentication Applications: Attacks, services, and Mechanisms; Security Attacks; Security Services; A model for Internetwork Security; Internet Standards and RFCs Kerberos, X.509 Directory Authentication Service.

UNIT 6**6 Hours**

Electronic Mail Security: Pretty Good Privacy (PGP); S/MIME

UNIT 7**6 Hours**

IP Security: IP Security Overview; IP Security Architecture; Authentication Header; Encapsulating Security Payload; Combining Security Associations; Key Management.

UNIT 8**6 Hours**

Web Security: Web security requirements; Secure Socket layer (SSL) and Transport layer Security (TLS); Secure Electronic Transaction (SET)

Text Books:

1. Michael E. Whitman and Herbert J. Mattord: Principles of Information Security, 2nd Edition, Cengage Learning, 2005. (Chapters 5, 6, 7, 8; Exclude the topics not mentioned in the syllabus)
2. William Stallings: Network Security Essentials: Applications and Standards, 3rd Edition, Pearson Education, 2007. (Chapters: 1, 4, 5, 6, 7, 8)

Reference Book:

1. Behrouz A. Forouzan: Cryptography and Network Security, Special Indian Edition, Tata McGraw-Hill, 2007.



SOFTWARE TESTING

Subject Code: 10CS842
Hours/Week: 4
Total Hours: 52

I.A. Marks: 25
Exam Marks: 100
Exam Hours: 3

PART – A

UNIT 1

6 Hours

A Perspective on Testing, Examples: Basic definitions, Test cases, Insights from a Venn diagram, Identifying test cases, Error and fault taxonomies, Levels of testing. Examples: Generalized pseudocode, The triangle problem, The NextDate function, The commission problem, The SATM (Simple Automatic Teller Machine) problem, The currency converter, Saturn windshield wiper.

UNIT 2

7 Hours

Boundary Value Testing, Equivalence Class Testing, Decision Table-Based Testing: Boundary value analysis, Robustness testing, Worst-case testing, Special value testing, Examples, Random testing, Equivalence classes, Equivalence test cases for the triangle problem, NextDate function, and the commission problem, Guidelines and observations. Decision tables, Test cases for the triangle problem, NextDate function, and the commission problem, Guidelines and observations.

UNIT 3

7 Hours

Path Testing, Data Flow Testing: DD paths, Test coverage metrics, Basis path testing, guidelines and observations. Definition-Use testing, Slice-based testing, Guidelines and observations.

UNIT 4

6 Hours

Levels of Testing, Integration Testing: Traditional view of testing levels, Alternative life-cycle models, The SATM system, Separating integration and system testing. A closer look at the SATM system, Decomposition-based, call graph-based, Path-based integrations.

PART – B

UNIT 5

7 Hours

System Testing, Interaction Testing: Threads, Basic concepts for requirements specification, Finding threads, Structural strategies and functional strategies for thread testing, SATM test threads, System testing guidelines, ASF (Atomic System Functions) testing example. Context of

interaction, A taxonomy of interactions, Interaction, composition, and determinism, Client/Server Testing,.

UNIT 6

7 Hours

Process Framework: Validation and verification, Degrees of freedom, Varieties of software. Basic principles: Sensitivity, redundancy, restriction, partition, visibility, Feedback. The quality process, Planning and monitoring, Quality goals, Dependability properties, Analysis, Testing, Improving the process, Organizational factors.

UNIT 7

6 Hours

Fault-Based Testing, Test Execution: Overview, Assumptions in fault-based testing, Mutation analysis, Fault-based adequacy criteria, Variations on mutation analysis. Test Execution: Overview, from test case specifications to test cases, Scaffolding, Generic versus specific scaffolding, Test oracles, Self-checks as oracles, Capture and replay.

UNIT 8

6 Hours

Planning and Monitoring the Process, Documenting Analysis and Test: Quality and process, Test and analysis strategies and plans, Risk planning, Monitoring the process, Improving the process, The quality team, Organizing documents, Test strategy document, Analysis and test plan, Test design specifications documents, Test and analysis reports.

TEXT BOOKS:

1. Paul C. Jorgensen: Software Testing, A Craftsman's Approach, 3rd Edition, Auerbach Publications, 2008.
(Listed topics only from Chapters 1, 2, 5, 6, 7, 9, 10, 12, 13, 14, 15)
2. Mauro Pezze, Michal Young: Software Testing and Analysis – Process, Principles and Techniques, Wiley India, 2009.
(Listed topics only from Chapters 2, 3, 4, 16, 17, 20, 24)

REFERENCE BOOKS:

1. Aditya P Mathur: Foundations of Software Testing, Pearson Education, 2008.
2. Srinivasan Desikan, Gopaldaswamy Ramesh: Software Testing Principles and Practices, 2nd Edition, Pearson Education, 2007.
3. Brian Marrick: The Craft of Software Testing, Pearson Education, 1995.

Clouds, Grids, and Clusters

Subject Code: 10CS845
Hours/Week: 4
Total Hours: 52

I.A. Marks: 25
Exam Marks: 100
Exam Hours: 3

PART – A

UNIT - 1 **6 Hours**

Introduction: Overview of Cloud Computing, Applications, Intranets and the Cloud, When can cloud Computing be used? Benefits and limitations, Security concerns, Regulatory issues

UNIT - 2 **6 Hours**

Business Case for Cloud, Examples of Cloud Services: Cloud computing services, Help to the business, Deleting the data center. Examples: Google, Microsoft, IBM, Salesforce.com and its uses, Cloud at Thomson Reuters.

UNIT - 3 **7 Hours**

Technology, Cloud Storage, Standards: Cloud Computing Technology: Clients, Security, Network, Services.

Overview of Cloud storage, Some providers of Cloud storage. Standards: Applications, Clients, Infrastructure, Service.

UNIT - 4 **7 Hours**

Other issues: Overview of SaaS (Software as a Service), Driving forces, Company offerings: Google, Microsoft, IBM. Software plus Service: Overview, Mobile device integration Local Clouds, Thin Clients, Migrating to the Cloud: Virtualization, Server solutions, Thin clients, Cloud services for individuals, mid-markets, and enterprises, Migration.

PART - B

UNIT - 5 **7 Hours**

GRID Computing – 1: Introduction: Data Center, The Grid and the Distributed/ High Performance Computing, Cluster Computing and Grid Computing, Metacomputing – the Precursor of Grid Computing, Scientific, Business and e-Governance Grids, Web services and Grid Computing, Business Computing and the Grid – a Potential Win win Situation, e-Governance and the Grid. Technologies and Architectures for Grid Computing: Clustering and Grid Computing, Issues in Data Grids, Key Functional Requirements in Grid Computing, Standards for Grid Computing , Recent Technological Trends in Large Data Grids. OGSA and WSRF: OGSA for Resource Distribution, Stateful Web Services in OGSA, WSRF (Web

Services Resource Framework), Resource Approach to Stateful Services, WSRF Specification.

The Grid and the Database: Issues in Database Integration with the Grid, The Requirements of a Grid enabled database, Storage Request Broker (SRB), How to integrate the Database with the Grid? The Architecture of OGSA-DAI for Offering Grid Database Services

UNIT - 6

6 Hours

GRID Computing – 2: World Wide Grid Computing Activities, Organizations and Projects: Standards Organizations, Organizations Developing Grid Computing Tool Kits, Framework and Middleware, Grid Projects and Organizations Building and Using Grid Based Solutions. Web Services and the Service Oriented Architecture (SOA): History and Background, Service Oriented Architecture, How a Web Service Works, SOAP and WSDL, Description, Creating Web Services, Server Side. Globus Toolkit: History of Globus Toolkit, Versions of Globus Toolkit, Applications of GT4 – cases, GT4 – Approaches and Benefits, Infrastructure Management, Monitoring and Discovery, Security, Data, Choreography and Coordination, Main Features of GT4 Functionality – a Summary, GT4 Architecture, GT4 Command Line Programs, GT4 Containers.

UNIT - 7

7 Hours

Cluster Computing – 1: Introduction: What is Cluster Computing, Approaches to Parallel Computing, How to Achieve Low Cost Parallel Computing through Clusters, Definition and Architecture of a Cluster, What is the Functionality a Cluster can offer? Categories of Clusters Cluster Middleware: Levels and Layers of Single System Image (SSI), Cluster Middleware Design Objectives, Resource Management and Scheduling, Cluster Programming Environment and Tools. Early Cluster Architectures and High Throughput Computing Clusters: Early Cluster Architectures, High Throughput Computing Clusters, Condor. Setting up and Administering a Cluster: How to set up a Simple Cluster? Design considerations for the Front End of a Cluster, Setting up nodes, Clusters of Clusters or Metacusters, System Monitoring, Directory Services inside the Clusters & DCE, Global Clocks Sync, Administering heterogeneous Clusters.

UNIT - 8

6 Hours

Cluster Computing – 2: Cluster Technology for High Availability: Highly Available Clusters, High Availability Parallel Computing, Mission Critical (or Business Critical or Business Continuity) Applications, Types of Failures and Errors, Cluster Architectures and Configurations for High Availability, Faults and Error Detection, Failure Recovery, Failover / Recovery Clusters. Performance Model and Simulation: Performance Measures and Metrics, Profit Effectiveness of Parallel Computing through Clusters. Process Scheduling, Load Sharing and Load Balancing: Job Management System

(JMS) Resource Management System (RMS), Queues, Hosts, Resources, Jobs and Policies, Policies for Resource Utilization, Scheduling Policies Load Sharing and Load Balancing, Strategies for Load Balancing, Modeling Parameters Case Studies of Cluster Systems: Beowulf, PARAM.

Text Books:

1. Anthony T. Velte, Toby J. Velte, Robert Elsenpeter: Cloud Computing, A Practical Approach, McGraw Hill, 2010.
2. Prabhu: Grid and Cluster Computing, PHI, 2008.

Reference Books:

1. Joshy Joseph, Craig Fellenstein: Grid Computing, Pearson Education, 2007.
2. Internet Resources

CLOUD COMPUTING [As per Choice Based Credit System (CBCS) scheme] (Effective from the academic year 2016 -2017) SEMESTER – II			
Subject Code	16SCS12/16SCE12 16SIT22/16SSE254 16SCN22/16LNI151	IA Marks	20
Number of Lecture Hours/Week	04	Exam Marks	80
Total Number of Lecture Hours	50	Exam Hours	03
CREDITS – 04			
Course objectives: This course will enable students to			
<ul style="list-style-type: none"> • Define and Cloud, models and Services. • Compare and contrast programming for cloud and their applications • Explain virtualization, Task Scheduling algorithms. • Apply ZooKeeper, Map-Reduce concept to applications. 			
Module 1			Teaching Hours
Introduction, Cloud Infrastructure: Cloud computing, Cloud computing delivery models and services, Ethical issues, Cloud vulnerabilities, Cloud computing at Amazon, Cloud computing the Google perspective, Microsoft Windows Azure and online services, Open-source software platforms for private clouds, Cloud storage diversity and vendor lock-in, Energy use and ecological impact, Service level agreements, User experience and software licensing. Exercises and problems.			10 Hours
Module 2			10 Hours
Cloud Computing: Application Paradigms: Challenges of cloud computing, Architectural styles of cloud computing, Workflows: Coordination of multiple activities, Coordination based on a state machine model: The Zookeeper, The Map Reduce programming model, A case study: The Goo The Web application, Cloud for science and engineering, High-performance computing on a cloud, Cloud computing for Biology research, Social computing, digital content and cloud computing.			
Module 3			10 Hours
Cloud Resource Virtualization: Virtualization, Layering and virtualization, Virtual machine monitors, Virtual Machines, Performance and Security Isolation, Full virtualization and paravirtualization, Hardware support for virtualization, Case Study: Xen a VMM based paravirtualization, Optimization of network virtualization, vBlades, Performance comparison of virtual machines, The dark side of virtualization, Exercises and problems.			
Module 4			10 Hours
Cloud Resource Management and Scheduling: Policies and mechanisms for resource management, Application of control theory to task scheduling on a cloud, Stability of a two-level resource allocation architecture, Feedback control based on dynamic thresholds, Coordination of specialized autonomic performance managers, A utility-based model for cloud-based Web services, Resourcing bundling: Combinatorial auctions for cloud resources, Scheduling algorithms for computing clouds, Fair queuing, Start-time fair queuing, Borrowed virtual time, Cloud scheduling subject to deadlines, Scheduling MapReduce applications subject to deadlines, Resource management and dynamic scaling, Exercises and problems.			
Module 5			10 Hours
Cloud Security, Cloud Application Development: Cloud security risks, Security: The top concern for cloud users, Privacy and privacy impact assessment, Trust, Operating system security, Virtual machine Security, Security of virtualization, Security risks posed by shared images, Security risks posed by a management OS, A trusted virtual machine monitor, Amazon web services: EC2 instances, Connecting clients to cloud instances through firewalls, Security rules for application and transport layer protocols in EC2,			

How to launch an EC2 Linux instance and connect to it, How to use S3 in java, Cloud-based simulation of a distributed trust algorithm, A trust management service, A cloud service for adaptive data streaming, Cloud based optimal FPGA synthesis. Exercises and problems.	
Course Outcomes:	
The students should be able to:	
<ul style="list-style-type: none"> • Compare the strengths and limitations of cloud computing • Identify the architecture, infrastructure and delivery models of cloud computing • Apply suitable virtualization concept. • Choose the appropriate cloud playwr • Address the core issues of cloud computing such as security, privacy and interoperability • Design Cloud Services • Set a private cloud 	
Question paper pattern:	
The question paper will have ten questions.	
There will be 2 questions from each module.	
Each question will have questions covering all the topics under a module. The students will have to answer 5 full questions, selecting one full question from each module.	
Text Books:	
1. Dan C Marinsecu: Cloud Computing Theory and Practice, Elsevier(MK), 2013.	
Reference Books:	
1. Rajkumar Buyya , James Broberg, Andrzej Goscinski: Cloud Computing Principles and Paradigms, Wiley 2014.	
2. John W Rittinghouse, James F Ransome:Cloud Computing Implementation, Management and Security, CRC Press 2013.	

MANAGING BIG DATA [As per Choice Based Credit System (CBCS) scheme] (Effective from the academic year 2016 -2017) SEMESTER - II			
Subject Code	16LN422 / 16SCE21 / 16SCN24 / 16SCS21 / 16SIT41 / 16SSE422	IA Marks	20
Number of Lecture Hours/Week	04	Exam Marks	80
Total Number of Lecture Hours	50	Exam Hours	03
CREDITS – 04			
Course objectives: This course will enable students to <ul style="list-style-type: none"> • Define big data for business intelligence • Analyze business case studies for big data analytics • Explain managing of Big data Without SQL • Develop map-reduce analytics using Hadoop and related tools 			
Module -1			Teaching Hours
UNDERSTANDING BIG DATA: What is big data – why big data – Data, Data Storage and Analysis, Comparison with Other Systems, Rational Database Management System , Grid Computing, Volunteer Computing, convergence of key trends – unstructured data – industry examples of big data – web analytics – big data and marketing – fraud and big data – risk and big data – credit risk management – big data and algorithmic trading – big data and healthcare – big data in medicine – advertising and big data – big data technologies – introduction to Hadoop – open source technologies – cloud and big data – mobile business intelligence – Crowd sourcing analytics – inter and trans firewall analytics.			10Hours
Module -2			10 Hours
NOSQL DATA MANAGEMENT: Introduction to NoSQL – aggregate data models – aggregates – key-value and document data models – relationships – graph databases – schema less databases – materialized views – distribution models – sharding – version – map reduce – partitioning and combining – composing map-reduce calculations.			
Module -3			10 Hours
BASICS OF HADOOP: Data format – analyzing data with Hadoop – scaling out – Hadoop streaming – Hadoop pipes – design of Hadoop distributed file system (HDFS) – HDFS concepts – Java interface – data flow – Hadoop I/O – data integrity – compression – serialization – Avro – file-based data structures.			
Module-4			10 Hours
MAPREDUCE APPLICATIONS: MapReduce workflows – unit tests with MRUnit – test data and local tests – anatomy of MapReduce job run – classic Map-reduce – YARN – failures in classic Map-reduce and YARN – job scheduling – shuffle and sort – task execution – MapReduce types – input formats – output formats			
Module-5			10 Hours
HADOOP RELATED TOOLS: Hbase – data model and implementations – Hbase clients – Hbase examples –praxis. Cassandra – Cassandra data model – Cassandra examples – Cassandra clients –Hadoop integration. Pig – Grunt – pig data model – Pig Latin – developing and testing Pig Latin scripts. Hive – data types and file formats – HiveQL data definition – HiveQL data manipulation – HiveQL queries.			
Course outcomes:			
The students shall able to: <ul style="list-style-type: none"> • Describe big data and use cases from selected business domains • Explain NoSQL big data management • Install, configure, and run Hadoop and HDFS 			

- Choose key Internet applications and their protocols, and apply to develop their own applications (e.g. Client Server applications, Web Services) using the sockets API.
- Explain develop effective communication mechanisms using techniques like connection establishment, queuing theory, recovery Etc.
- Explain various congestion control techniques.

Question paper pattern:

The question paper will have ten questions.

There will be 2 questions from each module.

Each question will have questions covering all the topics under a module. The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Larry Peterson and Bruce S Davis "Computer Networks :A System Approach" 5th Edition , Elsevier -2014.
2. Douglas E Comer, "Internetworking with TCP/IP, Principles, Protocols and Architecture" 6th Edition, PHI – 2014.

Reference Books:

1. Uyless Black, "Computer Networks, Protocols , Standards and Interfaces" 2nd Edition - PHI
2. Behrouz A Forouzan, "TCP/IP Protocol Suite" 4th Edition – Tata McGraw-Hill.

ADVANCES IN COMPUTER NETWORKS [As per Choice Based Credit System (CBCS) scheme] (Effective from the academic year 2016 -2017)			
SEMESTER – II			
Subject Code	16SCN12/16SCS22	LA Marks	20
Number of Lecture Hours/Week	04	Exam Marks	80
Total Number of Lecture Hours	50	Exam Hours	03
CREDITS – 04			
Course objectives: This course will enable students to			
<ul style="list-style-type: none"> • Discuss with the basics of Computer Networks. • Compare various Network architectures. • Discuss fundamental protocols. • Define and analyze network traffic, congestion, controlling and resource allocation. 			
Module 1			Teaching Hours
Foundation: Building a Network, Requirements, Perspectives, Scalable Connectivity, Cost-Effective Resource sharing, Support for Common Services, Manageability, Protocol layering, Performance, Bandwidth and Latency, Delay X Bandwidth Product, Perspectives on Connecting, Classes of Links, Reliable Transmission, Stop-and-Wait , Sliding Window, Concurrent Logical Channels. T1: Chapter 1.1, 1.2, 1.5.1, 1.5.2, 2.1, 2.5 T2: Chapter 4			10 Hours
Module 2			10 Hours
Internetworking I: Switching and Bridging, Datagram's, Virtual Circuit Switching, Source Routing, Bridges and LAN Switches, Basic Internetworking (IP), What is an Internetwork?, Service Model, Global Addresses, Datagram Forwarding in IP, sub netting and classless addressing, Address Translation (ARP), Host Configuration (DHCP), Error Reporting (ICMP), Virtual Networks and Tunnels. T1: Chapter 3.1, 3.2.			
Module 3			10 Hours
Internetworking- II: Network as a Graph, Distance Vector (RIP), Link State (OSPF), Metrics, The Global Internet, Routing Areas, Routing among Autonomous systems (BGP), IP Version 6 (IPv6), Mobility and Mobile IP T1: Chapter 3.3, 4.1.1, 4.1.3 T2: Chapter 13.1 to 13.18, Ch 18.			
Module 4			10 Hours
End-to-End Protocols: Simple Demultiplexer (UDP), Reliable Byte Stream(TCP), End-to-End Issues, Segment Format, Connecting Establishment and Termination, Sliding Window Revisited, Triggering Transmission, Adaptive Retransmission, Record Boundaries, TCP Extensions, Queuing Disciplines, FIFO, Fair Queuing, TCP Congestion Control, Additive Increase/ Multiplicative Decrease, Slow Start, Fast Retransmit and Fast Recovery T1: Chapter 5.1, 5.2.1 to 5.2.8, 6.2, 6.3			
Module 5			10 Hours
Congestion Control and Resource Allocation Congestion-Avoidance Mechanisms, DEC bit, Random Early Detection (RED), Source-Based Congestion Avoidance, The Domain Name System (DNS), Electronic Mail (SMTP,POP,IMAP,IMME), World Wide Web (HTTP), Network Management (SNMP) T1: Chapter 6.4 T2: Chapter 23.1 to 23.16, Chapter 24, Chapter 25, Chapter 27.1 to 27.8			
Course Outcomes:			
The students should be able to:			

- Choose key Internet applications and their protocols, and apply to develop their own applications (e.g. Client Server applications, Web Services) using the sockets API.
- Explain develop effective communication mechanisms using techniques like connection establishment, queuing theory, recovery Etc.
- Explain various congestion control techniques.

Question paper pattern:

The question paper will have ten questions.

There will be 2 questions from each module.

Each question will have questions covering all the topics under a module. The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Larry Peterson and Bruce S Davis "Computer Networks :A System Approach" 5th Edition , Elsevier -2014.
2. Douglas E Comer, "Internetworking with TCP/IP, Principles, Protocols and Architecture" 6th Edition, PHL - 2014.

Reference Books:

1. Uylas Black, "Computer Networks, Protocols , Standards and Interfaces" 2nd Edition - PHL
2. Behrouz A Forouzan, "TCP/IP Protocol Suite" 4th Edition - Tata McGraw-Hill.

INTERNET OF THINGS [As per Choice Based Credit System (CBCS) scheme] (Effective from the academic year 2016 -2017) SEMESTER – II			
Subject Code	16LNE233 /16SCE233 /16SCNI31 /16SCS24 /16SIT231 /16SSE421	IA Marks	20
Number of Lecture Hours/Week	04	Exam Marks	80
Total Number of Lecture Hours	50	Exam Hours	03
CREDITS – 04			
Course objectives: This course will enable students to <ul style="list-style-type: none"> • Define and explain basic issues, policy and challenges in the IoT • Illustrate Mechanism and Key Technologies in IoT • Explain the Standard of the IoT • Explain resources in the IoT and deploy of resources into business • Demonstrate data analytics for IoT 			
Module -1			Teaching Hours:
What is The Internet of Things? Overview and Motivations, Examples of Applications, IPv6 Role, Areas of Development and Standardization, Scope of the Present Investigation, Internet of Things Definitions and frameworks-IoT Definitions, IoT Frameworks, Basic Nodal Capabilities, Internet of Things Application Examples- Overview, Smart Metering/Advanced Metering Infrastructure-Health/Body Area Networks, City Automation, Automotive Applications, Home Automation, Smart Cards, Tracking, Over-The-Air-Passive Surveillance/Ring of Steel, Control Application Examples, Myriad Other Applications.			10Hours

Fundamental IoT Mechanism and Key Technologies-Identification of IoT Object and Services, Structural Aspects of the IoT, Key IoT Technologies, Evolving IoT Standards- Overview and Approaches, IETF IPv6 Routing Protocol for RPL Roll, Constrained Application Protocol, Representational State Transfer, ETSI M2M, Third Generation Partnership Project Service Requirements for Machine-Type Communications, CENELEC, IETF IPv6 Over Lowpower WPAN, Zigbee IP(ZIP), IPSO	10 Hours
Module -3	
Layer 2 Connectivity: Wireless Technologies for the IoT-WPAN Technologies for IoT/M2M, Cellular and Mobile Network Technologies for IoT/M2M, Layer 3 Connectivity -IPv6 Technologies for the IoT: Overview and Motivations, Address Capabilities, IPv6 Protocol Overview, IPv6 Tunneling, IPsec in IPv6, Header Compression Schemes, Quality of Service in IPv6, Migration Strategies to IPv6.	10 Hours
Module-4	
Case Studies illustrating IoT Design-Introduction, Home Automation, Cities, Environment, Agriculture, Productivity Applications.	10 Hours
Module-5	
Data Analytics for IoT – Introduction, Apache Hadoop, Using Hadoop MapReduce for Batch Data Analysis, Apache Oozie, Apache Spark, Apache Storm, Using Apache Storm for Real-time Data Analysis, Structural Health Monitoring Case Study.	10 Hours
Course outcomes: At the end of this course the students will be able to: <ul style="list-style-type: none"> • Develop schemes for the applications of IOT in real time scenarios • Manage the Internet resources • Model the Internet of things to business • Understand the practical knowledge through different case studies • Understand data sets received through IoT devices and tools used for analysis 	
Question paper pattern: The question paper will have ten questions. There will be 2 questions from each module. Each question will have questions covering all the topics under a module. The students will have to answer 5 full questions, selecting one full question from each module.	
Text Books: 1. Daniel Minoli, "Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications", Wiley, 2013. 2. Arundheep Balga, Vijay Madisetti, "Internet of Things: A Hands on Approach" Universities Press, 2013	
Reference Books: 1. Michael Miller, "The Internet of Things", First Edition, Pearson, 2015. 2. Claire Rowland, Elizabeth Goodman et al., "Designing Connected Products", First Edition, O'Reilly, 2013.	

INFORMATION AND NETWORK SECURITY [As per Choice Based Credit System (CBCS) scheme] (Effective from the academic year 2016 -2017) SEMESTER – II			
Subject Code	16LN11/16SCN13/16SCS253	LA Marks	20
Number of Lecture Hours/Week	03	Exam Marks	80
Total Number of Lecture Hours	40	Exam Hours	03
CREDITS – 03			
Course objectives: This course will enable students to			
<ul style="list-style-type: none"> • Explain standard algorithms used to provide confidentiality, integrity and authenticity. • Distinguish key distribution and management schemes. • Deploy encryption techniques to secure data in transit across data networks • Implement security applications in the field of information technology 			
Module 1			Teaching Hours
Classical Encryption Techniques: Symmetric Cipher Model, Cryptography, Cryptanalysis and Brute-Force Attack, Substitution Techniques, Caesar Cipher, Mono-alphabetic Cipher, Playfair Cipher, Hill Cipher, Poly alphabetic Cipher, One Time Pad. Block Ciphers and the data encryption standard: Traditional block Cipher structure, stream Ciphers and block Ciphers, Motivation for the feistel Cipher structure, the feistel Cipher, The data encryption standard, DES encryption, DES decryption, A DES example, results, the avalanche effect, the strength of DES, the use of 56-Bit Keys, the nature of the DES algorithm, timing attacks, Block cipher design principles, number of rounds, design of function F, key schedule algorithm.			8 Hours
Module 2			Teaching Hours
Public-Key Cryptography and RSA: Principles of public-key cryptosystems. Public-key cryptosystems. Applications for public-key cryptosystems, requirements for public-key cryptosystems. Public-key cryptanalysis. The RSA algorithm, description of the algorithm, computational aspects, the security of RSA. Other Public-Key Cryptosystems: Diffie-hellman key exchange, The algorithm, key exchange protocols, man in the middle attack, Elgamal Cryptographic systems, Elliptic curve arithmetic, abelian groups, elliptic curves over real numbers, elliptic curves over \mathbb{Z}_p , elliptic curves over $\text{GF}(2^m)$, Elliptic curve cryptography, Analog of Diffie-hellman key exchange, Elliptic curve encryption/ decryption, security of Elliptic curve cryptography, Pseudorandom number generation based on an asymmetric cipher, PRNG based on RSA.			8 Hours
Module 3			Teaching Hours
Key Management and Distribution: Symmetric key distribution using Symmetric encryption, A key distribution scenario, Hierarchical key control, session key lifetime, a transparent key control scheme, Decentralized key control, controlling key usage, Symmetric key distribution using asymmetric encryption, simple secret key distribution, secret key distribution with confidentiality and authentication, A hybrid scheme, distribution of public keys, public announcement of public keys, publicly available directory, public key authority, public keys certificates, X-509 certificates. Certificates, X-509 version 3, public key infrastructure. User Authentication: Remote user Authentication principles, Mutual Authentication, one way Authentication, remote user Authentication using Symmetric encryption, Mutual Authentication, one way Authentication, Kerberos, Motivation, Kerberos version 4, Kerberos version 5, Remote user Authentication using Asymmetric encryption, Mutual Authentication, one way Authentication, federated identity management, identity management, identity federation, personal identity verification.			8 Hours

<p>network measures, mobile device security, security threats, mobile device security strategy, IEEE 802.11 Wireless LAN overview, the Wi-Fi alliance, IEEE 802 protocol architecture. Security, IEEE 802.11i services, IEEE 802.11i phases of operation, discovery phase, Authentication phase, key management phase, protected data transfer phase, the IEEE 802.11i pseudorandom function. Web Security Considerations: Web Security Threats, Web Traffic Security Approaches. Secure Sockets Layer: SSL Architecture, SSL Record Protocol, Change Cipher Spec Protocol, Alert Protocol, and shake Protocol. Cryptographic Computations: Transport Layer Security: Version Number, Message Authentication Code, Pseudorandom Functions, Alert Codes, Cipher Suites, Client Certificate Types, Certificate Verify and Finished Messages, Cryptographic Computations, and Padding. HTTPS Connection Initiation, Connection Closure. Secure Shell(SSH) Transport Layer Protocol, User Authentication Protocol, Connection Protocol</p>	
<p>Module 5</p> <p>Electronic Mail Security: Pretty good privacy, notation, operational, description, S/MIME, RFC5322, Multipurpose internet mail extensions, S/MIME functionality, S/MIME messages, S/MIME certificate processing, enhanced security services, Domain keys identified mail, internet mail architecture, E-Mail threats, DKIM strategy, DKIM functional flow. IP Security: IP Security overview, applications of IPsec, benefits of IPsec, Routing applications, IPsec documents, IPsec services, transport and tunnel modes, IP Security policy, Security associations, Security associations database, Security policy database, IP traffic processing, Encapsulating Security payload, ESP format, encryption and authentication algorithms, Padding, Anti replay service, transport and tunnel modes, combining security associations, authentication plus confidentiality, basic combinations of security associations, internet key exchange, key determinations protocol, header and payload formats, cryptographic suits.</p>	<p>8 Hours</p>
<p>Course Outcomes</p> <p>The students should be able to:</p> <ul style="list-style-type: none"> Analyze the vulnerabilities in any computing system and hence be able to design a security solution. Identify the security issues in the network and resolve it. Evaluate security mechanisms using rigorous approaches, including theoretical. 	
<p>Question paper pattern:</p> <p>The question paper will have ten questions. There will be 2 questions from each module. Each question will have questions covering all the topics under a module. The students will have to answer 5 full questions, selecting one full question from each module.</p>	
<p>Text Books:</p> <p>1. William Stallings, Cryptography and Network Security, Pearson 6th edition.</p>	
<p>Reference Books:</p> <p>1. V K Pachghare: Cryptography and Information Security.</p>	