



B.Sc IT

(BACHELOR OF SCIENCE IN INFORMATION TECHNOLOGY)

GAUHATI UNIVERSITY

REVISED PROGRAM STRUCTURE AND DETAILED CURRICULUM

IN THE L-T-P-C FORMAT

YEAR 2014

COURSE CURRICULUM FOR B.Sc.(I.T)

(Bachelor of Science in Information Technology)

Rules and Regulations

1. The course will be a 3year (6 Semester) course.

2. **Eligibility:** Higher Secondary in Science with Mathematics securing the minimum pass mark in the subject.

3. Marks Distribution

i) Each theory paper will consist of 75 marks with 20% (i.e. 15 marks) for internal assessment and 80% (i.e. 60 marks) for written examination to be held at the end of the semester conducted by the University.

ii) There will be a practical paper in each semester. Mark break-up for practical papers will be as follows:

a) 20% as internal evaluation

b) 80% will be assigned in a practical Examination to be held at the end of the semester by the University in the presence of external examiner.

iii) Total mark for each practical paper has been given in the detailed syllabus.

iv) There will be a project in the 6th semester with a total mark of 150, out of which 30 marks will be internal and 120 marks will be based on project work and a viva-voce to be held at the end of the 6th semester by the University. The viva-voce will be held in the presence of an external examiner appointed by the University. The internal mark for project will be based on two project progress reports and corresponding presentation with an equal duration between the presentations. A project can be done individually or may be shared between two students.

4. A student completing the first two semesters successfully is eligible to get a certificate “**Certificate in Information Technology**”. A student completing first four semesters successfully is eligible to get a “**Diploma in Information Technology**” and students completing all the six semester successfully will get “**B.Sc. In Information Technology**” certificate.

5. All other matters related to examinations including internal evaluations will be as per the regulation of TDC for semester system of G.U. with choice based credit and grading system.

6. Project Rules:

-based on IT related topics

-there will be a project report

-the project should be a work done equivalent to 200 hours of effort.

-Project is appropriate for B. Sc. (I.T.) level course.

7. The **L-T-P-C** break- up for the courses has been mentioned below with each course. Each theory paper will carry a total of 6 credits except the course 6.4 project work which will carry 12 credits. For the theory papers,

1 credit point corresponds to one lecture / tutorial per week and for practical papers 1 credit point corresponds to 1 practical session (of at least 2 hours) per week.

Requirement for Colleges introducing B.Sc(IT) course

1. Number and Qualification of Teachers:

- i. At least 3 teachers having qualification either M.Sc(IT/CS), M.Tech (CS/IT), MCA from a recognized university having UGC norms.
- ii. One Laboratory-cum-Office Assistant: Any graduate in Science with either PG Diploma in CS/IT, or B. Sc(with major in CS), or BCA, or B,Sc(IT), or DOEACC-A level pass, or Diploma in Computer Engineering from Govt/AICTE/UGC recognized Institute.
- iii. One Office Bearer: 10 + 2 pass.

2. Resources required for opening the course in the college with an intake of 30 students:

- i) A laboratory with 30 computers
- ii) Server
- iii) Printer, Scanner, UPS, Networking equipments and room preparation (AC, vacuum cleaner) with Internet connection.
- iv) Furniture as required.
- v) Library books, journals etc.

COURSE BREAK-UP

SEMESTER – I

- 1.1 Communicative English (5-1-0-6)
- 1.2. Mathematics –I (5-1-0-6)
- 1.3 Digital logic (5-1-0-6)
- 1.4 Computer fundamental and Programming (5-1-0-6)
- 1.5. Practical (0-0-4-4)

SEMESTER – II

- 2.1. Mathematics – II (5-1-0-6)
- 2.2. ICT Hardware (4-2-0-6)
- 2.3. Data structure and algorithm (5-1-0-6)
- 2.4. Operating Systems (5-1-0-6)
- 2.5 Practical (0-0-6-6)

SEMESTER – III

- 3.1 Environmental Studies (5-1-0-6)
- 3.2 Computer Organization (5-1-0-6)
- 3.3 Object Oriented Programming (5-1-0-6)
- 3.4 Database Management System (5-1-0-6)
- 3.5 Practical (0-0-6-6)

SEMESTER – IV

- 4.1 Programming in Java (5-1-0-6)
- 4.2 Software Engineering (5-1-0-6)
- 4.3 Data Communication and Computer Networks (5-1-0-6)
- 4.4 Theoretical Foundation of Computing (5-1-0-6)
- 4.5. Practical (0-0-6-6)

SEMESTER – V

- 5.1 Compiler Design (5-1-0-6)
- 5.2. Web Technology (5-1-0-6)
- 5.3. Computer Graphics (5-1-0-6)
- 5.4. Microprocessor (5-1-0-6)
- 5.5. Practical (0-0-6-6)

SEMESTER –VI

- 6.1 System Administration (5-1-0-6)
- 6.2 Electives (5-1-0-6)
 - a) Data Mining and Warehousing
 - b) Artificial Intelligence
 - c) Computer Oriented Optimization Techniques
 - d) Object Oriented Analysis and Design
 - e) System Analysis and Design
 - f) Computer Oriented Numerical and Statistical Methods.
- 6.3 Practical (0-0-4-4)
- 6.4 Project (0-0-12-12)

1.1 COMMUNICATIVE ENGLISH

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

This course is designed to introduce students to aspects of effective communication, both oral and written. While the five units of the course are expected to acquaint I.T students with various levels of communication necessary in everyday life, the emphasis throughout is on helping them acquire the basic skills – particularly the ability to write and speak plain and correct English.

Fundamentals

Marks 10 [6 Hours]

It deal with the fundamentals of effective communication:

- i) the key stages in the Communication cycle,
- ii) the importance of cross-cultural communications,
- iii) the barriers to communication and
- iv) steps to effective communication.

Students will be required to answer questions on these issues.

Written Communications

Marks-20 [10 Hours]

Fundamentals of effective writing. Students will here be required to learn some of the basic patterns of English speech. Use of tenses: present, past and future. Modals (can, must, should). Articles and nouns. Pronouns and determiners. Relative clauses. Adjectives and adverbs. Conjunctions and prepositions. Phrasal verbs. Avoidance of clichés and common errors. Subject and verb agreement. Punctuation. Paragraphing

Oral Communications

Marks-15 [12Hours]

Oral communication in the workplace. The two-way nature of communication. Speaking skills / Listening skills. Group discussion and interviews. Effective speech (pronunciation / delivery)

Business Communication

Marks-15 [12Hours]

Different categories of business letters (confirmation, acknowledgment, enquiry, reply, circulars etc.). E-mail. Intercultural sensitivities.

Suggested Reading:

1. Raymond Murphy, *Murphy's English Grammar* (Cambridge University Press, 2004)
2. Michael Swan, *Practical English Usage* (Oxford University Press, 2001)
3. Shirley Taylor, : *A Communication for Business Practical Approach* (Pearson Education, 2006)
4. Bovee, Thill and Schatzman, *Business Communication Today* (Pearson Education, 2007)
5. *Cambridge International Dictionary of English* (Cambridge University Press, 2007)

1.2 MATHEMATICS-I

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Unit-1: Sequence and Series

Marks 15 [12 Hours]

Sequence of real numbers, bounded, convergent and non-convergent sequences. Uniqueness of the limit and bounds of a convergent sequence. Cauchy sequence, Cauchy's general principle of convergence (proof of the necessary part only). Sub sequences, convergence and divergence of monotonic sequences. Algebraic operations on limit (statements of the theorems without proof). Infinite series, statements of basic properties of infinite series (without proof). Absolute and conditional convergence. Tests for convergence : Comparison test, Ratio test.

Unit-2: Trigonometry

Marks 10 [8 Hours]

Geometrical representation of complex numbers the Argand plane. Polar form of a complex number. Modulus, amplitude and their various properties. De Moivre's theorem. Expansion of $\cos(x)$ and $\sin(x)$ in positive integral powers of x . Gregory's series, Hyperbolic functions.

Unit-3 : Abstract Algebra

Marks 15+5=20 [20 Hours]

Group Theory :

Definition and examples of groups. Per mutation group and cyclic group. Subgroups and Cosets Lagrange's theorem on the order of a subgroup of a finite group. Normal subgroups. Quotient groups. Homomorphism of Group; properties of Homomorphism ; isomorphism of group.

Ring Theory :

Definition and examples. Simple properties of Rings. Integral domain, Field and their elementary properties.

Unit-4:

Mark 5 [10 Hours]

Relation between the roots and Coefficients of a general Polynomial equation in one variable. Transformation of equations Descartes' rule of signs, Symmetric functions of roots; Solution of Cubic equation by Cardon's Methods.

Unit-5: Calculus

Marks 10 [10 Hours]

Roll's theorem, Lagrange's Mean Value theorem and Taylor's theorem. Meaning of the sign of derivative. Indeterminate forms, maxima and minima (single variable).

Suggested Readings :

1. Higher Algebra (Classical); S. K. Mapa; Ashok Prakashan, Kolkata.
2. Higher Trigonometry, Das and Mukherjee, U. N. Dhur and Sons.
3. A course in Abstract Algebra; V. K. Khanna & K.S. K. Bhambri, Vikas Pub. House, Pvt. Ltd., New Delhi.
4. Modern Algebra; S. Singh and Q. Zameer ruddin; Vikas Pub. House, Pvt. Ltd., New Delhi.

1.3 DIGITAL LOGIC

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Introduction to Digital Logic

Marks 13 [12 Hours]

Unit1: Boolean algebra and Logic gates

Boolean operators, axiomatic definition of Boolean algebra, Rules (postulates and basic theorems) of Boolean algebra, dual and complement of Boolean expression, Canonical form and Standard form, Sum of product and product of sum for m. Conversion between Boolean expression and truth table. Boolean expression and their simplification by algebraic method, Karnaugh map method (till four variable k-map) and Quine Mc Cluskey method, Don't care condition.

Logic gates:

Different types of gates, Implementation of logic expression with logic gates.

Unit2: Combinational circuit

Marks 13 [12 Hours]

Adder: half adder, full adder, parallel binary adder, Subtractors: half subtractor and full subtractor, Magnitude comparator, Decoder, Encoder, Application examples of decoder and encoder, Multiplexer, Demultiplexer, Application examples of multiplexer and demultiplexer, programmable logic Array (PLA)

Unit3: Sequential Circuit

Marks 12 [10 Hours]

Simple RS flipflop or latch, Clocked RS flipflop, D flipflop, JK flipflop, T flipflop, Edge triggered flipflop (SR, D, JK), Asynchronous preset and clear inputs, master Slave Flip Flop, JK Master slave flip flop., edge triggering and level triggering. Analysis of Clocked Sequential circuits, State Reduction and Assignment, Flip-Flop Excitation tables. Design Procedure for sequential circuits.

Unit4: Counters

Marks 10 [13 Hours]

Ripple counters: Binary Ripple Counter, BCD Ripple Counter. Synchronous Counters: Binary Counter, Binary Up and down Counter, BCD Counter, Timing Sequences. Ring counter and Johnson counter. Counter design using state diagram, state table and state equation.

Unit5: Registers and the Memory Unit

Marks 12 [13 Hours]

Registers: Shift registers (serial in serial out, serial in parallel out, parallel in serial out, parallel in parallel out), Registers with parallel Load, Bidirectional shift register with parallel load.

Memory Unit: Semiconductor memory: RAM, RAM Family, Examples of RAM: Integrated circuit Memory, Magnetic core Memory, Cache memory. Flash memory, Basic flash memory, Comparison of flash memory with other memories, SIMM and DIMM.

Suggested Reading

1. Digital Logic and Computer Design, Morris M. Mano
2. Digital Fundamentals, Floyd and Jain

1.4 COMPUTER FUNDAMENTALS AND PROGRAMMING

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Fundamentals

Marks :15

Major components of a Digital Computer (A brief introduction of CPU, Main memory, Secondary memory devices and I/O devices) Keyboard, monitor, mouse, printers, Secondary storage devices (floppy disks, hard disks and optical disks), backup system and why it is needed ? Bootstrapping a Computer. Representation of numbers (only a brief introduction to be given) and characters in computer. ASCII. EDCDIC and Gray codes. Interpreter, Assembler, Linker and Loader. Definition and concepts of algorithm and its different implementations-pseudo code, flowchart and Computer programs.

Number System: Binary, Hexadecimal, Octal, BCD, and conversions of number systems. Representation of signed integers, Sign and magnitude, 1's complement and 2's complement representation. Arithmetic operations using 2's complement representation and conditions for overflow/underflow and its detection.

Introduction to C

Marks : 25

Elementary data types , variables, constants and identifiers. Integer, character floating point and string constants . variable declarations. Syntax and semantics. Reserved word. Initialization of variable during declarations Constant data types. Expression in C, precedence and associativity of C operators, unary, binary and ternary operators. C arithmetic operators, assignment operators, relational operators, logical operators and bit –wise operators . L-value and R-value. Side effects of operators. Expression statement. Conditional Statement-if, if-else, switch

Iterative Statement-while, do-while, for

Other Statement –break , continue, goto, return, null Statement, block Statement. Function: function declaration. Calling a function. Parameters –Call by value, Call by reference and its absence in C. Recursion and how it works. Cast and sizeof operator. Automatic type Conversion.

Simple programs like programs to compute an arithmetic expression, unit conversion, the sum of a series (like trigonometric series), GCD, factorial (both recursive and non-recursive version), fibonacci number (both recursive and non-recursive version), generation of prime numbers, reversing digits of an integer, finding the square root of a number, prime factors of an integer, base conversion of numbers, test if three points form a triangle and classify triangles as right angled, isosceles, equilateral etc., roots of a quadratic equation, generation of simple patterns of characters on screen.

Arrays and pointers

Marks : 15

Storage classes : Automatic, External, Static, Register. Scope and lifetime of variables. Arrays and pointers and corresponding operators. Pointer arithmetic. Programs using arrays and pointers like sum, average, minimum, maximum of an array of numbers. Add and delete an element of an array. Merge

two sorted arrays. String manipulation programs like string concatenation, palindrome, reverse, copy etc. Matrix manipulation like Sum of rows, columns, and diagonal elements of a matrix, transpose of a matrix.

Searching and Sorting

Linear search, binary search. Selection sort and bubble sort.

Structures and Files:

Marks 5

Structure – declaration and use. Structure member resolution and structure pointer member resolution operators. Programs to show the use of structure. Standard C library.

Files in C—opening, closing, reading and writing of files. Seeking forward and backward. Simple examples of file handling programs.

Suggested Reading:

Programming with C, B.S. Gottfried, Tata Mc-Graw Hill.
Programming in ANSI C, E.Balagurusamy, Tata McGraw – Hill
The C Programming Language, B.W. Kernighan and D.M.Ritchie, PHI
Computer Fundamentals, Anita Goel, Pearson, 2010.

1.5 PRACTICAL

Total Marks:50 (Internal mark 10, semester end examination 40)

At least 20 practical assignments covering paper 1.4 should be done by the students from the list prepared by the UG Committee of Courses and Studies in Computer Science. The assignments are to be selected in such a way that the whole course is covered.

2.1 MATHEMATICS II

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Unit1: Sets, Relations and Functions

Marks 10[12 Hours]

Sets, relations, properties of binary relations, closures of relation, equivalence relations, equivalence classes and partitions. Partial ordering relations and lattices. Functions, one-to-one and onto, principles of mathematical induction.

Unit2: Graph Theory

Marks 13 [10 Hours]

Basic Definition of graph. Connectivity of graph, cut points cycles, Hamiltonian graphs, trees, different Characterization of trees, bipartite graph, Algorithms on graph, Breadth first search, Depth first search.

Unit3: Combinatorics

Marks 8 [Hours 8]

Basic of counting principles, principle of inclusion exclusion, application of inclusion and exclusion. Pigeonhole principle, generalized Pigeonhole principle and its application, permutations and combinations, permutations with repetitions, combinations with repetitions, permutations of sets with indistinguishable objects.

Unit4: Matrices

Marks 12 [15 Hours]

Row and column operations, vectors and matrices, partitioning of matrices, representing relations using matrices, Determinant of a square matrix, minor, cofactor, the Cayley-Hamilton theorem, inverse of a matrix, product form of inverse. Rank of a matrix. Solutions of simultaneous linear equations, existence of solutions, solution by Gaussian elimination, Eigen values and Eigen vectors.

Unit5: Logic

Marks 10 [15 Hours]

Connectives, truth tables, normal forms CNF, DNF, Converting expressions to CNF and DNF, Theory of inference, Propositional calculus. Boolean Algebra. Predicate calculus (only introduction), predicates and quantifiers.

Unit6: Vector Space

Marks 7 [10 Hours]

Definition and examples of vector spaces. Elementary properties of \mathbb{R} as a vector space. Subspaces of a vector space. Union, intersection and sum of two subspaces. Subspaces generated by a subset of a vector space. Definition, example and properties of linearly independent and dependent set of vectors. Basis and dimension of a vector space. Examples of finite dimensional vector spaces.

Suggested Readings :

1. Discrete Mathematical Structure, Kolman /Rahman Pearson Education.
2. Discrete Mathematics and its Applications, K. H. Rosen, Mc-Graw Hill International Ed.
3. Discrete Mathematics structures with applications to Computer Science, J. P. Tremblay and R. Manohar, Mc-Graw Hill.
4. Discrete Mathematics, N. Ch. S.N. Iyengar, K. A. Venkatesh, V. M. Chandrasekaran, P. S. Arunachalam, Vikash Publishing House Pvt. Ltd.

2.2 ICT HARDWARE

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

UNIT I: Evolution of computer system, Modern computer, Classification of computer, Personal Computer hardware: Monitor, Keyboard, Mouse, Scanner, printer, speaker **12 marks**

UNIT II: Hard Disk Drive: logical structure and file system, FAT, NTFS. Hard disk tools: Disk cleanup, error checking, de fragmentation, scanning for virus, formatting, installing additional HDD. New trends in HDD. Floppy Disk Drive **12 marks**

UNIT III: Optical Media, CDROM, theory of operation, drive speed, buffer, cache, CD-r, CD-RW, DVD ROM, DVD technology, preventive maintenance for DVD and CD drives, New Technologies. Driver installation, Writing and cleaning CD and DVD. **12 marks**

UNIT IV: Processor: Intel processor family. Latest trends in processor, Motherboard, Sockets and slots, power connectors. Peripheral connectors. Bus slots, USB, pin connectors. Different kinds of motherboards. RAM, different kinds of RAM. RAM up gradation. Cache and Virtual Memory concept. **12 marks**

UNIT V: SMPS. BIOS. Network Interface Card, network cabling, I/O Box, Switches, RJ 45 connectors, Patch panel, Patch cord, racks, IP address. **12 marks**

BOOKS RECOMMENDED :

1. Comdex: Hardware and Networking Course Kit:: DreamTech press
PC hardware: A beginners Guide: Ron Gilster: Tata Mc Graw Hill

2.3 DATA STRUCTURE AND ALGORITHM

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Fundamentals

Marks : 10

Concept of Data Types, elementary structure, words and their interpretations, packed words

Arrays : Types, memory representation, address translation functions for one & two dimensional arrays, different examples.

Linked Structure:

Marks : 10

Singly and doubly linked list, circular and non circular, list manipulation with pointers , example involving insertion and deletion of elements and their comparative studies with implementations using array structure

UNIT 3 :

Marks : 10

Stacks and Queues

Definitions, representation using array and linked list structure, application of stack and queues in simulation, postfix conversion and evolution of arithmetic expressions

UNIT 4:

Marks : 10

Binary Trees:

Definition, quantitative properties, memory representation, Trees traversal algorithms (recursive and non-recursive), threaded trees.

BFS and DFS

Sorting and Searching :

Marks :15

Linear and binary search algorithms, performance and complexity, binary search trees (construction ,insertion , deletion and search) Concept of optimal binary search trees. Terminology, performance evaluation, sorting algorithms (non recursive, recursive description, Complexity, advantages and disadvantage, implementation)

Bubble sort, insertion sort, selection sort, heap sort, quick sort, merge sort & radix sort. External Sorting.

Analysis of Algorithm

Marks: 5

Time and Space complexity of algorithms, average case and worst case analysis, asymptotic notation as a measure of algorithm complexity, O and notations. Analysis of sorting algorithms- Selection sort, Bubble sort, Insertion sort, Heap sort, Quick sort and analysis of searching algorithms – linear search and binary search.

Suggested Readings:

1. Data Structure , Horowitz and Sahani, Narosa
2. Introduction to Data Structures in C, A.N.Kamthane, Pearson, 2007.
3. Data Structure using C and C++, Langsam, Augentein&Tanenbaum, PHI
4. Data Structures using C, S.K.Bandyopadhyay, K.N.Dey, Pearson.

2.4 OPERATING SYSTEMS

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Unit1: Introduction

Marks 5 [4 Hours]

What is an operating system, evolution of operating systems. simple batch systems, time-sharing systems, personal computer systems, parallel systems, distributed systems, real time systems. functions of an operating systems.

Unit2 : Memory Management

Marks 10 [12 Hours]

logical versus physical address space, swapping. contiguous allocation, paging, segmentation, fragmentation, segmentation with paging. protection.

Virtual Memory: paging, demand paging, page replacement algorithms, , page tables, TLBs-translation lookaside buffers, inverted page tables, thrashing.

Unit3: Processes and Threads

Marks 17 [16 Hours]

Processes: Process model, process creation, process termination, Process hierarchies, process states , implementation of processes.

Threads: The thread model, thread usage, threads in user space and kernel, threads in linux

Interprocess Communication: race conditions, critical regions, mutual exclusion with busy waiting, sleep and wakeup, semaphores, message passing, mail box, interprocess communication in linux.

Classical IPC problems: the dining philosophers problem, the readers and writers problem, the sleeping barber problem.

Scheduling: introduction to scheduling, scheduling in batch system and interactive systems, scheduling in real time systems, policy versus mechanism, thread scheduling.

Unit4: Deadlocks

Marks 18 [16 Hours]

System model, conditions for deadlock , conditions for deadlock modeling, deadlock detection with one and multiple resource of each type ,recovery from deadlock , deadlock avoidance, deadlock prevention . **Secondary storage Structure:** Disk structure, disk scheduling, disk management, disk reliability .

Device Management: Techniques for device management, dedicated devices, shared devices, storage devices, buffering,

Input/Output:

Principles of I/O hardware: I/O devices, device controller, memory mapped I/O, direct memory access

Principle of I/O software: programmed I/O, interrupt driven I/O, I/O using DMA.

I/O software layers: Interrupt handlers, device drivers, device independent I/O software, Userspace I/O software.

Disks: Disk formatting, disk arm scheduling algorithm.

Unit5: File System**Marks 10 [12Hours]**

File Concept, file structure, file types, file access, file attributes, file operation, directory structure, directory operations, linux file system ,file system structure, implementing files, implementing directories, shared files ,file system reliability, file system performance, allocation methods, freespace management. file system calls in linux. Structure of ext2 file system. ext3 file system and journalling.

Suggested reading:

1. Tanenbaum “Modern Operating Systems”
2. Silberschatz , Galvin, Gang , ‘Operating system concepts’

2.5 PRACTICAL

[Full Marks 75]

Laboratory-

Total Marks:75 (Internal mark 15, semester end examination 60)

At least 6 practical assignments covering paper 2.2 (15 marks), 15 assignments covering paper 2.3 (25 marks) and 12 assignments covering paper 2.4 (20 marks) should be done by the students from the list prepared by the UG Committee of Courses and Studies in Computer Science. The assignments are to be selected in such a way that the whole course is covered.

3.1 ENVIRONMENTAL STUDIES

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Unit I: The Multidisciplinary nature of environmental studies

4 lectures Marks 3

Definition, scope and importance

Need for public awareness

Unit 2: Natural Resources

8 lectures Marks 10

- Renewable and nonrenewable resources
- Natural resources and associated problems

a)Forest resources:

Use and over-exploitation, deforestation, case studies. Timber extraction, mining, dams and their effects on forest and tribal people.

b)Water resources:

Use and over utilization of surface and ground water, floods, drought, conflicts over water, damsbenefits and problems.

c)Mineral resources:

Use and exploitation, environmental effects of extracting and using mineral resources, case studies.

d)Food resources:

World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizerpesticide problem, water logging, salinity, case studies.

e)Energy resources:

Growing energy needs, renewable and nonrenewable energy sources, use of alternate energy sources, case studies.

f)Land resources:

Land as a resource, land degradation, man induced landslides, soil erosion and desertification.

Role of an individual in conservation of natural resources.

Equitable use of resources for sustainable lifestyles.

Unit 3: Ecosystems

8 lectures Marks 8

- Concept of an ecosystem
- Structure and function of and ecosystem.
- Producers, consumers and decomposers
- Energy flow in the system
- Ecological succession
- Food chains, food webs and ecological pyramids
- Introduction, types, character istic features, structure and function of the following ecosystem:
 - a) Forest ecosystem
 - b) Grassland ecosystem
 - c) Desert ecosystem
 - d) Aquatic ecosystem (ponds, streams, lakes, rivers, oceans, estuaries)

Unit 4: Biodiversity and its conservation

8 lectures Marks 8

Introduction definition, genetics, species and ecosystem diversities
Biogeographical classification of India

Value of biodiversity: consumptive use, productive use, social, ethical, aesthetic and option values.

Biodiversity at global, national and local level.

India as a megadiversity nation.

Hotspots of biodiversity

Threats to biodiversity: habitat loss, poaching of wild life, manwildlife conf licts.
Endangered and endemic species of India.

Conservation of biodiversity: Insitu and exsitu conservation of biodiversity

Unit 5: Environmental pollution

8 lectures Marks 7

- Definition
- Causes, effects and control measure of
 - a. air pollution
 - b. water pollution
 - c. soil pollution
 - d. marine pollution
 - e. noise pollution
 - f. thermal pollution
 - g. nuclear pollution
- Solid waste management:** Causes, effects and control measures of urban and industr ial waste.
- Role of individual in prevention of pollution

- Pollution case studies
- Disaster management: floods, earthquake, cyclone and landslides.

Unit 6: Social issues and the Environment:

8 lectures Marks 7

- From unsustainable to sustainable development
- Urban problems related to energy
- Water conservation, rain water harvesting, watershed management
- Resettlement and rehabilitation of people, its problem and concern, case studies.
- Environmental ethics: issues and possible solutions
- Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust, case studies.
- Waste land reclamation
- Consumerism and waste product
- Environment protection acts
- Air(prevention and control of pollution) acts
- Wild life protection act
- Forest conservation act
- Issues involved in enforcement of environmental legislation, public awareness.

Unit 7: Human population and the environment

8 lectures Marks 7

- Population growth, variation among nations
- Population explosion family welfare programme
- Environment and human health
- Human rights
- Value education
- HIV/ AIDS
- Women and child welfare
- Role of information technology in environment and human health
- Case studies

Unit 8: Field work

Marks 10

- Visit to a local area to document environmental assets
river/forest/grassland/hill/mountain
- Visit to a local polluted site urban/rural/ industrial/ agricultural
- Study of common plants, insects, birds
- Study of simple ecosystem, ponds, river, hill slopes etc.

3.2 COMPUTER ORGANIZATION

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Unit1: Introduction

Marks 10 [12 Hours]

Functional units of a computer, basic instructions, interconnection of functional units, bus structure, memory locations, memory addresses, memory operations, instruction and instruction sequencing (straight Line sequencing and branching), addressing modes, introduction to assembly language, stack, subroutine, I/O instructions.

Unit2: Register Transfer Logic

Marks 8 [8 Hours]

Introduction, inter register transfer, arithmetic microoperation, logic microoperation, shift microoperation, Conditional control statements, fixed point binary data, instruction code, design of a simple computer.

Unit3: Processor logic design

Marks 8 [8 Hours]

Processor organization, design of arithmetic and logic unit, status register, design of accumulator .

Unit4: Control logic design

Marks 8 [8 Hours]

Hardware control, microprogrammed control block diagram, symbolic microprogram, microprogrammed CPU organization.

Unit5: I/O Subsystem

Marks 13 [12 Hours]

Program controlled I/O, Interrupts: enabling and disabling interrupts, handling interrupts from multiple sources (priority control), DMA, structure and working of hard disk, CDROM, printer.

Unit6: Memory subsystem

Marks 13[12Hours]

Semiconductor memory, SRAM, DRAM, ROM, speed size and cost, Cache memory, Mapping functions, replacement algorithms.

Suggested Reading:

1. Computer System Architecture, M.Morris Mano, PHI publication
2. Computer Architecture, Hamachar, Vranesic and Zaky.

3.3 OBJECT ORIENTED PROGRAMMING

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Unit1: Principles of Object Oriented Programming

Marks 10 [8 Hours]

Basic concept of OOP, Procedural programming vs OOP. Introducing Object Oriented Approach relating to other paradigms. Benefits of OOP and methods. Applications of OOP. Abstraction, Encapsulation, Inheritance, Polymorphism, Review of C, Difference between C and C++

Unit2: Basic Elements and Ideas

Marks 10 [8 Hours]

Keywords and Identifiers in C++, Variables and Constants, Declaration and Initialization of Variables, Concept of Dynamic Initialization of variables, Enumerated variables, Basic data types, Arrays and Strings, User Defined Data types, Arithmetic, Relational, Logical Operators and Operator Precedence, Manipulators, Type Conversions and type cast operators, Console I/O: cin, cout functions. Control Statements.if; if-else; else...if; switch statements. Loops: for, while, do-while, Break, continue, go to. Pointer new, delete operators. Functions in C++: main function, function prototyping, call by reference, return by reference, functions inline, friend, virtual, library.

Unit3: Classes and Objects

Marks 10 [8 Hours]

Encapsulation, information hiding, abstract data types, Object & classes, attributes, functions, C++ class declaration, member functions, State identity and behavior of an object, static data members and member functions, friend functions, constant member functions. Constructors and destructors, instantiation of objects, Default parameter value, object types.

Unit4: Overloading

Marks 8 [8 Hours]

Function overloading: Function overloading with various data types, scoping rules for function overloading, Special features of function overloading. Operator overloading: Concept of Operator Overloading, overloading of Unary and Binary Operators., overloading binary operators using friends, manipulation of strings using operators.

Unit5: Inheritance:

Marks 8[8 Hours]

Concept of inheritance, Types of Inheritance Single, multilevel, multiple, hierarchical, hybrid, virtual base class, abstract class. Type of Derivations – public, private & protected, Constructors in Derived Classes.

Unit6: Virtual Functions and Polymorphism

Marks 5 [8 Hours]

Polymorphism, Categorization of polymorphism techniques: compile time polymorphism, Polymorphism by parameter, run time polymorphism pointers to derived class, Early binding vs. late binding, virtual function, pure virtual function.

Unit7:Files and Exception Handling

Marks 5[8 Hours]

Need For a Data File, Opening and Closing a File, Detecting Endof File, Classes for File Stream Operations, Sequential Input and Output Operations, Random Access File Processing. Concept and Use of Exception Handling. Throwing Mechanism, Catching Mechanism,Specifying Exception.

Function Template and Class Template.

Suggested readings:

- i) Herbert Schild, "The complete reference to C++", Osborn McGraw Hill
- ii) R. Lafore, "Object Oriented Programming using C++", Galgotia Publications
- iii) Ian Graham, "Object Oriented Methods", Addison Wesley.
- iv) E. Balaguruswamy, "Object Oriented Programming with C++", Tata McGraw Hill 1997.
- v) D. ravichandran, "Programming With C++", Tata McGraw Hill Publishing Company Ltd.

3.4 DATABASE MANAGEMENT SYSTEMS

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Unit1: File structure

Marks 8 [6 Hours]

Record storage and primary file organization: memory hierarchies and storage devices, Storage of Databases, Placing file records on disks: Records and its Types, Files, Fixed length records and variable length records, Record Blocking, allocating file blocks on disks, operation on files.

Issues in Physical Design: Concept of indexes

Unit2: Overview of Database Management System:

Marks 12 [10 Hours]

Definition of Database, Traditional File Approach vs. DBMS approach, Characteristics of The Database approach, DBMS user, Role of a DBA, Advantage of using DBMS, DBMS architecture, Data independence

ANSI/SPARC 3 level architecture.

Unit3: Relational Models

Marks 20 [17Hours]

Fundamental integrity rules: entity integrity, referential integrity, Relational algebra (Select , Project,Cross ,Product , theta join, equi join, natural join, outer join),Set Operation ANSI SQL –92Standard: DDL, DML, SQL constructs (Select .. From... Where... Group by Having ... Order by....), Insert, Delete, Update, View, Definition and use, nested quires, Constraints considers (NOT NULL , UNIQUE, Check Primary key. Foreign key)

Unit4: Database Design

Marks 20 [17Hours]

Conceptual model, logical model, physical model. ER model as a tool for conceptual design entities, attributesand relationships, weak andstrong entities, conversion of ER model into relational schema.

Normalization: informal design guidelines for relational schemas (overview level), functional dependencies, different types of keys. Normal forms (first, second, third, BCNF).

Suggested reading:

1. Introduction to database management system, C.J. Date
2. Fundamentals of data base management system, Elmasri & Navathe

3.5 PRACTICAL

Total Marks:75 (Internal mark 15, semester end examination 60)

At least 6 assignments covering paper 3.4 (35 marks) and 12 assignments covering paper 3.3 (25 marks) should be done by the students from the list prepared the UG Committee of Courses and Studies in Computer Science. The assignments are to be selected in such a way that the course is covered.

4.1 PROGRAMMING IN JAVA

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

UNIT-I

Marks : 30

Java language basics: Basic features, Java virtual machine concepts. Data types:- primitive data types and variables. Java Key words, integer and floating point data type, character and Boolean types, declaring and initialization variables. Java operators, Expressions, Statements-- selection statements, control statements, iterative statements, jump statements.

Classes & Objects: Creating objects, assigning object reference variables. Introducing methods.

Arrays:- Static methods, Constructors, Overloading constructors., Strings, and Vector.

This Keyword , Using object as parameters:- argument passing, returning objects.

Method Overloading , garbage collection , the Finalize() method.

Inheritance and Polymorphism- Inheritance basics, access control, multilevel inheritance, method overriding, abstract classes, polymorphism, Final keyword.

Multithreaded programming,

I/O in Java :- I/O basics, Streams and stream classes , reading from and writing to console, reading and writing files

UNIT – II

Marks : 8

Java applets: The Applet Class, Applet Architecture, An Applet skeleton: initialization and termination, handling events, HTML Applet Tag, Control.

UNIT – III

Marks : 7

Networking : Socket overview -- datagram socket and TCP/IP based server socket, Internet Addressing --- DNS, UR. Event handling Drivers in Java.

UNIT – IV

Marks : 15

Java Database Connectivity: Establishing a connection(JDBC, ODBC connectivity) , transactions with database.

Suggested Reading

1. Programming in Java, E. Balaguruswamy, Tata Mc-Graw Hill.
2. Patrik Naughton & Herbert Schildt “ The complete reference java”

4.2 SOFTWARE ENGINEERING

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Unit1 : Introduction

Marks 15 [12 Hours]

Software Processes & Characteristics, Software life cycle Models- Waterfall, Prototype, Evolutionary and Spiral Models Software Requirements analysis & specifications: Requirement engineering, requirement, elicitation techniques like FAST, QFD, requirements analysis using DFD, Data dictionaries, ER Diagrams, Requirements documentation, Nature of SRS, Characteristics & organization of SRS.

Unit2 : Software Project Planning

Marks 15 [16 Hours]

Size Estimation like lines of Code & Function Count, Cost Estimation Models, COCOMO, Risk Management.

Unit3: Software Design

Marks 15 [16 Hours]

Data design, Architectural design, Interface design, Function Oriented Design, Object Oriented Design, Cohesion & Coupling, Classification of Cohesiveness & Coupling, Software Metrics: different types of project metrics.

Unit4 : Software Testing and Maintenance

Marks 15 [16 Hours]

Testing Process, Design of Test Cases, Types of Testing, Functional Testing, Structural Testing, Test Activities,

Unit Testing, Integration Testing and System Testing. Debugging Activities Software Maintenance: Management of Maintenance, Maintenance Process, Reverse Engineering, Software Re engineering, Configuration Management, Documentation. Software quality Assurance.

CASE tools Analysis tools, design tools, SQA tools, software testing tools.

Suggested readings:

1. Rajeev Mall “Software Engineering” PHI
2. Pressman Roger “ Software Engineering A Practitioners Approach” Tata McGraw Hill
3. James F. Peters, Witold Pedrycz “Software Engineering An Engineering Approach

4.3 DATA COMMUNICATION AND COMPUTER NETWORKS

Full marks: 75 (Internal 15 marks)

INTRODUCTION

Marks:6

Usage of Computer Network, study of topology, concept of protocol, Connection less and connection Oriented Service, Layered architecture, study of OSI and TCP model.

PHYSICAL LAYER

Marks:10

Introduction to Guided and Unguided media, physical description of twisted pair, coaxial cable, and fiber optic cable, Maximum data rate of a channel (Nyquist and shannons law), Basic concepts of Modulation and demodulation, Data encoding techniques (Manchester and Differential Mancestar encoding) . Network connecting devices hub, repeater, bridge, switch, router, and gateway

DATA LINK LAYER

LLC

Marks: 7

Functions and services of DLL, Framing and Framing Methods, Concept of Error Control, Error Correcting code(Hamming code), Error detecting code(CRC), Concept of Flow Control, Piggybacking, Stop-and-Wait sliding window protocol, Pipelining techniques(Go backN, Selective Repeat).

MAC

Marks: 7

What is MAC? Static Channel Allocation, Dynamic Channel Allocation, Pure ALOHA, Slotted ALOHA, Carrier Sense Protocol, 1-persistent CSMA, Non-PersistentCSMA, CSMA/CD, Ethernet(IEEE 802.3) and Ethernet Frame Format, Basic concept of Wireless LAN(IEEE 802.11), Binary Exponential Backoff Algorithm.

NETWORK LAYER

Marks:10

Services and Functions of Network Layer, Virtual Circuit and Datagram Subnet, Routing, Distance Vector Routing, the Count-to-Infinity problem, Link State Routing, Congestion (definition and factors of congestion only), Definition of Quality of Service, Traffic shaping, Leaky Bucket and token Bucket Algorithm, Concept of IP Address.

TRANSPORT LAYER

Marks: 10

Functionality of transport Layer, Establishment and release of connection, TCP and UDP(Overview), Introduction to Sockets and socket primitives, port numbers.

APPLICATION LAYER

Marks: 10

Concept of E-mail, Telnet, WWW, DNS, HTTP, FTP, URL, SMTP, MIME.

Suggested Readings

- 1.Red Hat Linux:Proffitt:PHI
- 2.Introduction to system Administration:IBM series:PHI

4.4 THEORETICAL FOUNDATION OF COMPUTING

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Unit1: Finite Automata

Marks 10[10 Hours]

DFA, NFA, NFA with ϵ moves. Equivalence of DFA and NFA. Reduction of the number of states in a finite automata.

Unit2: Regular Languages and Regular Grammar

Marks 7 [6 Hours]

Concept of languages and grammar. Regular expressions. Connection between regular expressions and regular languages. Regular grammars, Right and Left Linear Grammars. Equivalence between Regular languages and Regular grammars.

Unit3: Properties of Regular Languages

Marks 10 [9 Hours]

Closure under simple set operations union, intersection, concatenation, complementation and star closure. proof of pumping lemma for regular language. Proof of nonregularity using Pigeonhole principle and using pumping lemma for regular languages.

Unit4: Context free languages

Marks 10[10 Hours]

Context free grammars, leftmost and rightmost derivations, derivation trees. Parsing and Ambiguity in grammars and languages. Simplification of Context free Grammars removing useless productions, empty productions and unit productions. Normal forms Chomsky and Greibach normal forms.

Unit5: Pushdown Automata

Marks 8 [10 Hours]

Definition and language accepted (acceptance by empty stack and final state and their equivalence). Pushdown Automata and Context free languages. Deterministic PDA and Deterministic Context free Languages.

Unit7: Properties of Context free Languages

Marks 5 [5 Hours]

Pumping Lemma for CFL. Using Pumping Lemma to show certain languages not to be Context free. Closure properties of CFL – closure under union, concatenation and star closure and showing that CFLs are not closed under intersection and complementation.

Unit8: Complexity Classes

Marks 10 [10 Hours]

Time and Space complexity of algorithms, average case and worst case analysis, asymptotic notation as a measure of algorithm complexity, O , Θ , Ω , o , ω , and notations. Analysis of sorting algorithms Selection sort, Bubble sort, Insertion sort, Heap sort, Quick sort and analysis of searching algorithms – linear search and binary search.

Suggested Reading

1. An introduction to Formal Languages and Automata, Peter Linz, Narosa.
2. Introduction to Automata Theory, Languages and Computation, Hopcroft and Ullman, Addison Wesley.
3. K. L. P. Mishra, N. Chandrasekaran; Theory of Computer Science (Automata, Languages and Computation), P. H. I.
4. T. H. Cormen, C. E. Leiserson and R. L. Rivest, Introduction to Algorithms, Tata Mcgraw Hill Publishers.

4.5 PRACTICAL

Total Marks:75 (Internal mark 15, semester end examination 60)

At least 8 practical assignments covering paper 4.1(30 marks), 3 assignments covering paper 4.2 (10 marks) and 4 assignments covering paper 4.3 (20 marks) should be done by the students from the list prepared the UG Committee of Courses and Studies in Computer Science. The assignments are to be selected in such a way that the course is covered.

5.1 COMPILER DESIGN

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Unit I : Introduction

Marks: 10[8 Hours]

What is a compiler? Phases of compiler. Overview of working of a compiler, linker, loader.

Unit II : Lexical Analysis

Marks: 13 [12 Hours]

NFA, DFA, conversion from NFA to DFA. Regular expression. Regular expression to NFA conversion. Minimization of DFA. ,Structuer of Lexical analyzer ,use of finite autometa to write lexical analyser .

Unit III : Syntax analysis

Marks: 12 [12 Hours]

Grammar representation. Derivation and parse tree. Ambiguity and possible elimination. Top down parsing. Recursive descent and predictive top down parsing. Elimination of Left recursion. Bottom up parsing. Operator precedence parsing, LR parsing (including SLR and LALR). Er ror detection and recovery. Parser table construction.

Unit IV : Code generation

Marks: 15 [16 Hours]

Symbol table contents, implementation. Type checking. Syntax directed translation. Forms of intermediate codes. Abstract Syntax Trees, Directed Acyclic Graph, Three address code. Intermediate code generation for different language constructs , boolean expressions, if, ifelse, while, case or switch. Target code generation issues, registerallocation, Runtime storage management

Unit V : Code Optimisation

Marks: 10 [12 Hours]

DAG, basic blocks, Common subexpression elimination, variable propogation, code motion, strength reduction, elimination of dead code, loop optimisation.

Suggested Reading:

- 1) Aho, Sethi, Ullman; Compilers, Principles, Techniques, Tools, Pearson Education
- 2) Compiler Design, Santanu Chattopadhyay, P.H.I.

5.2 WEB TECHNOLOGY

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Internet Basics: Marks: 3 [4 hours]

History of the internet , the world wide web, getting connected ,web page, home page, web site, Internet services: email.

Client Server Model: Marks: 15 [12 hours]

Structure of an HTML document. HTML tags. The HTTP protocol details. Client side software. Web browsers (Netscape/Mozilla as example). DHTML Web server architecture and functions of an web server. JDBC and ODBC. Server side vs client side scripts, advantages and disadvantages of each. Client side and server side scripting languages and their uses. Dynamic web page. CGI scripts, Java Scripts and JSP as examples. PHP and Perl as scripting language. Browser plugins.

Web Object Model: CORBA Marks: 5[4 hours]

XML: Marks: 12[10 hours]

Well formed XML syntax. references, well formed documents XML semantics. DTD, XML Schema, RELAXNG. Displaying XML on web. XML extensions. Processing XML files, Using a programming language and the SAX API,Using a programming language and the DOM API,Using a transformation engine and a filter ,Push Parsing ,Data binding ,Nonextractive XML

Distributed Multitiered Application: Marks: 20[12 hours]

J2EE Components ,J2EE Clients, Web Components, Business Components. J2EE Containers, Container Services , Container Types, Enterprise JavaBeans Technology, Java Servlet Technology, JavaServer Pages Technology.

Application server: Persistence, Transaction processing, Concurrency control, Events using, Java Message Service, naming and directory services (JNDI),Security (Java Cryptography Extension (JCE) and JAAS)
Deployment of software components in an application server
Remote procedure calls using RMI/IIOP. Exposing business methods as Web Services.

J2EE 1.4 API Marks: 15 [12 hours]

Java Message Service API, Java Transaction API, JavaMail API, JavaBeans Activation Framework, Java API for XML Processing, Java API for XMLBased RPC, SOAP with Attachments API for Java, Java API for XML, Registries, J2EE Connector Architecture, JDBC API

Web Security:**Marks: 5[3hours]**

Firewall, wrapper and Proxy.

Suggested reading:

1. The Internet Complete, M.L. Young
2. Using CGI by J. Dwight, M. Erwin, R. Niles.
3. Mastering JavaScript and Jscript by J. Jaworski
4. Dynamic HTML by D. Godmann.
5. Understanding HTML by D.P. Nagpal
6. <http://java.sun.com/j2ee>

5.3 COMPUTER GRAPHICS

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

UNIT 1:

Marks:15 [15 hours]

Introduction: computer graphics and its applications.

Graphics Devices:

Input Devices : Keyboard, Mouse, Trackball & Space ball, Joystick, Data Glove,

Digitizers, Image Scanners, Touch panels, Light Pens systems.

Output display devices : Refresh CRT, Rasterscan display and Randomscan display technique, color display techniques Beam penetration method and Shadowmask method, Direct view storage tubes, emissive & nonemissive flatpanel displays Plasma panels, Thinfilm electrostatic displays, LED and LCD monitor, Threedimensional viewing devices and VirtualReality systems;

Display processor :Rasterscan systems, Randomscan systems,

UNIT 2:

Marks:15 [15 hours]

Output primitives: linedrawing algorithms DDA algorithm and Bresenham's Line Algorithm, Mid point Algorithm for Circle and Ellipse Generation, Curve generation.

Attributes for output primitives : Areafilling Algorithms scanline polygonfill, Nonzerowinding number rule; Scanline curve filling, Boundaryfill algorithm, Flood fill algorithm; Character generation techniques generation of bitmap and outlined font.

UNIT 3 :

Marks:15 [15 hours]

2D Geometric Transformations: Basic transformation translation, rotation and Scaling, matrix representations and Homogeneous coordinate representations, Composite transformations among translation, rotation and scaling, General Pivotpoint rotation, General fixedpoint scaling, General scaling directions, Other transformations reflection and shear, Transformation between coordinate Systems, Definition of Affine transformations.

2D viewing: definition, viewing transformation pipeline, window to viewport coordinate transformation.

2D Clipping : Concept and Algorithm: point clipping, line clipping CohenSutherland algorithm, Area clipping, text clipping, polygon clipping.

Interactive picture construction techniques: Basic positioning methods, constraints, grids, gravity fields, rubberband methods, dragging, painting & rawing.

UNIT 4:

Marks:8 [10 hours]

3D concepts: Display methods Parallel projection, perspective projection,

3D geometric transformations: Transformation, Translation, Rotation and Scaling around axes, 3D Viewing Projections – Parallel and Perspective.

UNIT 5:

Marks:7 [5 hours]

Visible surface detection: Definition, Algorithms for visible surface detection – Depth buffer method, Abuffer method, Raycasting method, Curved surfaces, Wireframe Methods

Illumination and Surface rendering: definition and importance, light sources, Basic illumination models Ambient light, Diffuse ref lection, Specula reflector and Phong model, combined diffuse and secular ref lections for multiple light sources, Warn model, Intensity attenuation, Color considerations, Transparency, Shadows.

Suggested Readings:

1. Computer Graphics, D. Hearn and M.P.Baker, PHI Ltd.

5.4 MICROPROCESSOR

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Unit1: Internal Organization of 8085A microprocessor

Marks 10 [10 Hours]

User Programmable registers, PC, SP, accumulator, flags, data bus, address bus, control bus, instruction word size, opcode format, data format, memory addressing, I/O addressing, address decoding for memory and I/O.

Unit2: 8085A microprocessor architecture

Marks 15 [10 Hours]

Pinout of 8085A microprocessor, multiplexed address/data bus, control and status signal, demultiplexing of control signals, other signals, bus timings, fetch decode and execute cycle, timing diagram for opcode fetch memory read and memory write, interfacing memory and I/O.

Unit3: Assembly language programming in 8085A microprocessor

Marks 15[17 Hours]

Complete instruction set in detail, programming examples, logic operation, counters and time delays, stack and subroutine, processing arrays, bit manipulation.

Unit4: Interfacing

Marks 15 [14 Hours]

IN and OUT instruction, decoding addresses, Interfacing LED, relay, seven segment display, switch, keyboard,.

Unit5: Interrupts

Marks 5 [11 Hours]

Vectored interrupts, interrupt priorities, general purpose programmable peripheral devices, 8255A control and status registers, programming 8255A, introduction to 8279, 8254 and 8237 (block diagrams and basic functions).

Suggested Reading :

Microprocessor Architecture, Programming and Application with the 8085 by Ramesh S. Gaonkar

5.5 : PRACTICAL

Total Marks:75 (Internal mark 15, semester end examination 60)

At least 28 practical assignments covering paper 5.2 (35 marks) and 8 assignments covering paper 5.4 (25 marks) should be done by the students from the list prepared by the UG Committee of Courses and Studies in Computer Science. The assignments are to be selected in such a way that the whole course is covered.

6.1 SYSTEM ADMINISTRATION USING LINUX

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Unit I :

Marks:10

What is System Administration? Duties of a System Administrator. Basic features of the Linux operating system. Installation requirements, Partitioning the Hard drive in Linux, Installing the Linux system, Linux system Startup and Shutdown.

Unit II :

Marks: 14

Basics of Linux file system: hierarchy and types. commands *ls*, *cp*, *mv*, *rm*, *mkdir*, *rmdir*, *more*, *touch*, *ln*, *mount*, *umount*, *mkfs*., absolute and relative path names. Linux file types, attributes of file, setting user and group ownership of files and access permissions. mounting and unmounting file systems and partitions. Structure of */etc/fstab* file and its purpose. hard link, symbolic link. Introductions of Shells. Linux environment variables. Basics of Shell Programming

Unit III :

Marks: 14

Program and Process. Foreground process, Background Process, daemons. Basic commands for starting and stopping processes. Examining the list of running processes on the system and understand the data presented there. cron, crontab file format, Standard I/O, Standard error ,redirection and piping.

Unit IV :

Marks: 10

Managing user accounts: Adding a user,password,Creating Groups,adding and deleting groups,viewing user account information, structure of */etc/passwd*, */etc/shadow*, */etc/group* files System monitoring and logging, Monitoring memory usage, disk space usage. Backup and Restore procedure

Unit V:

Marks: 12

IP address and IP address classes, subnetting, CIDR, Interface configuring with *ifconfig*, Gateway configuration, adding routes, ping, netstat, traceroute. Understanding the significance of the */etc/services* file and well known port numbers. Server configuration DHCP, NFS, NIS, SAMBA, PROXY..

Suggested Readings

- 1.Red Hat Linux:Proffitt:PHI
- 2.Introduction to system Administration:IBM series:PHI
- 3.Essential System Administration:Frisch:O'REILLY

6.2(A) DATA MINING AND WAREHOUSING

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Data Warehousing:

Marks: 15[12 Hours]

Overview and concepts: Need for Data Warehousing, Basic elements of Data Warehousing, differences between Database Systems and Data Warehouse.

Planning and Requirements: Project planning and management, collecting the requirements.

Architecture and Infrastructure: Data Warehouse Architecture and its components, Infrastructure and metadata.

Data Design and Data Representation: Principles of dimensional modeling, advanced topics data extraction, transformation and loading, data quality.

Information Access and Delivery: Matching information to classes of users, OLAP in Data Warehouse, Data warehousing and the web.

Implementation and Maintenance: Physical design process, Data Warehouse deployment, growth and maintenance.

Data Mining

Introduction :

Marks: 10[5 Hours]

Basics of data mining, Different definitions of Data Mining and related concepts, Data mining process Data preparation, data cleaning and data visualization. KDD process.

Data mining techniques: Clustering, Association rules and Decision trees.

Clustering:

Marks: 12[12 Hours]

Partitional versus Hierarchical Clustering, types of data in clustering. Partitional clustering methods – kmeans, kmedoids, PAM, CLARA, CLARANS. Hierarchical clustering methods – BIRCH, CURE. Density based clustering methods DBSCAN. Categorical clustering – DBSCAN.

Rule Mining:

Marks: 12[13 Hours]

What is an association rule? Mining association rules, frequent sets and border sets, algorithms for mining association rules – Apriori algorithm, PincerSearch algorithm, Border algorithm. Generalized association rule, quantitative association rule, association rule with item constraint.

Decision Trees:

Marks: 6[5 Hours]

Introduction, tree construction principle, decision tree generation algorithms – CART, ID3.

Advanced Topics:**Marks: 5[8 Hours]**

(only basics of the following topics):

Web mining : Web Content Mining, Web Structure , Mining, Web Usage mining.Spatial mining, Temporal mining – Temporal association rules, sequence mining and GSP algorithm, discovery of frequent episodes.

Suggested Reading:

1. A.K. Puzari, Data Mining Techniques, University Press.
- 2.J. Han and M. Kamber. Data Mining: Concepts and Techniques. Morgan Kaufman. 2001.

6.2.(B) ARTIFICIAL INTELLIGENCE

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

UNIT: I

Mark:12[12 Hours]

Definition of artificial intelligence, Numerical computation, information storage, repetitive operations, other definitions of artificial intelligence, numeric versus symbolic, algorithm versus non algorithms, area of artificial intelligence, expert system, natural language processing, speech recognition, automatic programming, organization of artificial intelligence system, the underlying assumptions, artificial intelligence techniques.

UNIT : II

Marks: 12[13 Hours]

Is the good solution absolute or relative, production systems, production system characteristics, problem solving: defining the problem as a state space search, Water Jug Problem , Basic problem solving methods : Reason forward from the initial states , Reason backward from the goal states, Problem trees versus Problem Graphs, Knowledge representation: Matching and Indexing.

UNIT: III

Marks: 12[12 Hours]

Heuristic search, Heuristic functions, ORgraph, ANDOR graph, Weak methods: Generate and Test, Hill Climbing, Breadth first search, Best first search OR graph, Problem reduction, Constraints satisfaction, MeansEnd Analysis.

UNIT: IV

Marks: 12[13 Hours]

Game playing: The Minimax Procedure , Adding Alpha Beta Cutoffs, Knowledge Representation using predicate logic, Representing simple facts in logic, Augmenting the representation with computable functions and predicates, Resolution, Conversion to clause form, The basis of resolution, Resolution in propositional logic , The Unification algorithm, Resolution in predicate logic , Resolution algorithm for predicate logic, Introduction to Nonmonotonic Reasoning, Statistical and probabilistic reasoning.

UNIT: V

Marks: 12[10 Hours]

Natural language Understanding, Introduction to Understanding, What makes understanding hard, Understanding single sentences , Keyword matching , Syntactic analysis , semantic analysis , semantic grammars ,Case grammars Learning: Introduction to learning, Random learning and Neural nets, Learning by parameter adjustment, Learning in General Problem Solver (GPS), Concept Learning.

Suggested readings:

- 1.Artificial Intelligence Elaine Rich McGraw Hill book Co. 1982.
- 2.Artificial Intelligence PH Winston, Addison Wesley,1983.

3. Artificial Intelligence Concepts, Techniques and Applications Yoshikai Shirai & Junichi Tsujii, John Wileysons.

4. Artificial Intelligence A knowledge based application MW Richaugh ,PWS Rent Publishing, Boston.

6.2.(C) COMPUTER ORIENTED OPTIMIZATION TECHNIQUES

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Unit I : Linear Programming Techniques **Marks 15[Hours 12]**

The simplex algorithm, Charma's method of penalties, the two phase algorithm, problem of degeneracy and cycling, Duality theorem, revised simplex algorithm, revised simplex method versus simplex method. Sensitivity analysis, changes in the requirement vector, the cost vector and the coefficient matrix.

Unit II :Transportation and Assignment problem **Marks 15 [Hours 16]**

Various algorithms such as the algorithm of stepping stones, Hungarian etc.

Unit III : Nonlinear programming **Marks 15 [Hours 12]**

Constraint minima and maxima, necessary and sufficient condition for maxima and minima: KuhnTucker principle, quadratic programming

Unit IV : Queuing theory **Marks 8 [Hours 12]**

The exponential distribution, queue disciplines such as M/M/1, M/M/C.

Unit V : Simulation **Marks 7 [Hours 8]**

Even type of simulation, Monte Carlo techniques, simulation techniques applied to queues.

Suggested Readings:

1. S.L Gass, "Linear Programming"
2. K.V.Mittal & G Mohan, " Optimization Methods"
3. K.Swarap, P.K.Gupta, M.Mohan, " Operation Research"

6.2.(D) OBJECT ORIENTED ANALYSIS AND DESIGN

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Introduction: **Marks 5[2 Hours]**

What is object oriented programming? Usefulness of object oriented development.

Object Modeling: **Marks 10[8 Hours]**

Objects and classes, Links and Association, advanced links and association concepts, Generalization and Inheritance, grouping constructs, Aggregations, Abstract Classes, Generalization as Extension and Restriction.

Dynamic Modeling: **Marks 7[7 Hours]**

Events and States, Operations, Nested State Diagram, Concurrency, Synchronization of concurrent activities, relation of objects and dynamic models.

Function Modeling: **Marks 8[8 Hours]**

Data Flow Diagram, Process, Data Flow, Actor, Data Store, Control Flow, Operations and Constraints, Relation of functional to object and dynamic model.

Design Methodology: **Marks 15[15 Hours]**

Analysis, Problem statement, Design: Breaking a system into a sub systems, Identifying Concurrency, allocating subsystems to processors and tasks, Management of data Stores, Handling Global Resources, Handling Boundary Conditions, Choosing Software Control Implementation, setting trade off priorities.

Object Design: **Marks 15[20 Hours]**

Overview of object design, Combining the Three Models, Design Optimization, Implementation of Control, Adjustment of inheritance, design of association, Object Representation, Physical Packaging and Documentation.

Recommended Books:

Object Oriented Modeling and Design, James Rumbaugh, Blaha, Premelani. Eddy, Lorensen

6.2.(E) SYSTEM ANALYSIS AND DESIGN

Total marks: 75 (Semester end examination - 60, Internal assessment - 15)

Introduction:

Marks : 5[6 Hours]

System Definition and Concepts: General Theory systems, Manual and automated Systems, Real life Business Sub –Systems. System Environments and Boundaries .Real Time and distributed systems. Basic principles of successful systems. Approach to system Development: Structured system Analysis and Design, Prototype, joint Application Development.

Systems Analyst:

Marks : 5[2 Hours]

Role and Need of Systems Analyst. Qualification and responsibilities .System Analysis as a profession

System Development Cycle:

Marks : 8[6 Hours]

Introduction to Systems Development Life Cycle(SDLC) Various phases of SDLC: Study, Analysis, Design, Development, Implementation, Maintenance Systems documentation consideration: principles of systems Documentation, Types of documentation and their importance, Enforcing documentation discipline in an organization.

System Planning:

Marks : 8[10 Hours]

Data and fact gathering techniques: interviews, Group Communication Questionnaires, presentations & Site visits. Assessing project Feasibility: Technical, Operational, Economic, Cost Benefits Analysis, Schedule Legal and contractual, political. Modern methods for determining system requirements: joint Application, Development program, prototyping, Business process Reengineering. System Selection plan and proposal.

Modular and Structured Design:

Marks: 5[3 Hours]

Module Specifications. Top –Down and bottom- up design. Module coupling and Cohesion. Structure charts.

System design and Modeling:

Marks : 10[15 Hours]

Process Modeling, logical and physical design, Conceptual Data Modeling : entity Relationship Analysis, Entity –relationship Modeling, ERDs and DFDS Concepts of Normalization. Process Description: Structured English, Decision Tree, Decision Tables Documentation: Data Dictionary Recording Data Description

Input and output:**Marks : 7[9 Hours]**

Classification of forms, input/ output forms design. User interface design, Graphical interfaces. Standards and guidelines for GUI Design. Designing physical Files and databases: Designing Fields, Designing Physical Records , Designing Physical Files, Designing Databases. Introduction to CASE Tools, Features Advantages and Limitations of CASE Tools, Awareness about some commercial CASE Tools.

System Implementation and Maintenance:**Marks : 7[5 Hours]**

Planning Consideration. Conversion methods, procedures and controls. System acceptance criteria. System Evaluation and performance. Testing and Validation. Preparing User Manual. Maintenance Activities and Issues.

Computer System Audit and Security:**Marks : 5[3 Hours]**

Audit of Computer System Usage. Types of Threats to Computer System and Control Measures: Threat and Risk Analysis, Disaster Recovery and Contingency planning, viruses.

Recommended Books:

1. J Hoffer, "Modern Systems Analysis and Design", Joey George and Joseph Valacich, Pearson Education
- 2 A Dennis and BH Wixom, "System Analysis and Design" John Wiley and Sons, Inc.

6.2(F) COMPUTER ORIENTED NUMERICAL AND STATISTICAL METHODS

Total marks: 75 (Semester end examination -60, Internal assessment - 15)

[Although there is no practical examination for this paper, the students will have to write and execute computer programs for the numerical techniques studied in the course and for the internal evaluation the teachers are advised to consider this work done by the students]

Representation of numbers: **Marks: 8**
Floating point representation, single and double precision, round off errors and truncation errors.

Solution of non-linear equation: **Marks: 8**
Bisection method, Newtons method, Regula Falsi method, secant method, fixed point algorithm.

Solution of simultaneous linear equation: **Marks: 8**
Basic elimination method, Gaussian elimination method, Gauss Jordan method, method of successive approximation.

Ordinary differential equation: **Marks: 8**
Euler's method, Runge Kutta method, Milnes method.

Interpolation: **Marks: 8**
Newton's interpolation, Lagrange's interpolation, Newton's divided difference method.

Numerical integration: **Marks:8**
Trapezoidal rule, Simpson rule, Newton's Cotes method.

Statistical methods: **Marks: 12**

Measure of central tendency: Mean, Median and Mode.

Probability, probability distribution, Binomial, Poisson and normal distribution.

Mathematical expectations, moments, correlation, regression.

Suggested Reading:

- 1.M.K.Jain, S.R.K.Iyenger, R.K.Jain, " Numerical methods for Scientific and Engineering Computation", Wiley Easterns.
2. K.E. Atkinson, " An introduction to numerical analysis", J.Willey and Sons.

6.3 PRACTICAL

Total Marks:50 (Internal mark 10, semester end examination 40)

At least 15 practical assignments covering paper 6.1 should be done by the students from the list prepared the UG Committee of Courses and Studies in Computer Science. The assignments are to be selected in such a way that the whole course is covered.

6.4 PROJECT WORK

Total marks: 150(Internal assessment - 30)

Each student will be assigned some project work at the starting of the sixth semester. The objective of the project is to train the student to independently search, identify and study real-life important topics in CS/IT; to develop skills among students in a particular field of CS/IT; and to expose students to the world of technology, innovation, and research. Each student (or group of at most 2 students) is expected to take a unique problem under the guidance/supervision of a faculty member of the department. The problem should be such that the students get a chance to explore one or two technologies in depth and grab good command over those technologies after successful completion of the project. Repetition of the problems already attempted by students of the previous years should not be encouraged unless the problem has exceptionally great research importance and scope. Application problems, if found interesting and arisen at the demand of a particular situation, may also be assigned; but typical information management systems with just two or three simple database tables and/or data-entry forms are to be discouraged. The project may be done in other Institutes/Organizations with prior permission from the concerned department of the College and in this case also one project supervisor should have to be from the concerned department in the College. The work will have to be submitted in the form of a dissertation. Project presentation and evaluation will have to be done as per the regulation of TDC for semester system of G.U. with choice based credit and grading system.

PRACTICAL ASSIGNMENT LIST FOR Bsc(IT)

(Bachelor of Information Technology)

1.5 LABORATORY

At least 20 programming assignments have to be done by each student from the following list. The assignments should be selected in such a way that all the features of C language are included.

1. Write a program to convert a given temperature value from Fahrenheit scale to Centigrade scale and vice versa.
2. Write a program to display ASCII value of a character.
3. Write a program to check whether a number is perfect or not.
4. Write a program to find out the biggest of three numbers using nested if.
5. A company insures its drivers if either of the following conditions are satisfied
 - Driver is married.
 - Driver is an unmarried, male and above 30 years of age.
 - Driver is unmarried, female and above 25 years of age.Write a program to decide if a driver is to be insured using logical operators.
6. Write a program to read a list of positive integers terminated by -1 and display the odd and even numbers separately and also their respective counts.
7. Write a program to read values of n and x and print the value of y using switch case where
 - a. $y=n+x$ when $n=1$
 - b. $y=1+x/n$ when $n=2$
 - c. $y= n+3x$ when $n=3$
 - d. $y=1+nx$ when $n>3$ or $n<1$.
8. Write a program to n values of sales and then calculate the commission on sales amount where the commission is calculated as follows:
 - a. If sales \leq Rs.500, commission is 5%.
 - b. If sales > 500 but ≤ 2000 , commission is Rs 35 plus 10% above Rs 500.
 - c. If sales > 2000 but ≤ 5000 , commission is Rs 185 plus 12% above Rs.2000.
 - d. If sales > 5000 , commission is 12.5%.
9. Write a program to find out minimum, maximum, sum and average of n numbers without using array.
10. Program to find mean and standard deviation (SD) for a set of n numbers without using array.
11. Write a program to find out the roots of a quadratic equation. Use proper testing to find checks for real and complex roots.
12. Write a program to print the digits of a number in words. (eg. if a number 841 is entered through the keyboard your program should print "Eight Four One".)
13. Write a program to print the PASCAL Triangle up to the n-th row where n is an input to the program.
14. Write a function to return the HCF of two positive integers. Write a main function to read two positive integers and print their HCF and LCM by using the above function.
15. Write a program to convert a decimal number into binary number using function.
16. Write a program to display the result of sine series using function.

17. Write a program to find the sum of the following series

$$1+x-x^3/3!+x^5/5!-x^7/7!+\dots \text{corrected up to the 3 decimal place.}$$

18. Write a program to read n numbers in a sorted array and insert a given element in a particular position

19. Write functions to compute the factorial of a number using both recursive and non-recursive procedure.

20. Write a program to print the values of ${}^n C_r$ and ${}^n P_r$ for given positive integers $n \geq r > 0$. Use a function fact(n) to return the factorial of a non-negative integer.n.

$${}^n C_r = n! / r! * (n-r)! \quad {}^n P_r = n! / (n-r)!$$

21. Write a program to display the first n Fibonacci numbers using function.

22. Write a program to display the prime numbers within a given range. Write a function to check whether a given integer is prime or not and use it.

23. Write a program to Multiply two matrices using function

24. Write a program to display the upper Triangle and lower Triangle of a given square matrix using function.

25. Write a function to check if a given square matrix is symmetric or not. Write a main function to implement it.

26. Write a program to read a m X n matrix and calculate the Row sum and Column Sum of the matrix

27. Write a function to read in an integer and print the representation of the number using the sign and magnitude representation scheme using 8 bits. The program should check for overflow/under flow conditions. The left most bit is to be used as the sign bit.

28. Write a program to merge two sorted arrays.

29. Write a program to implement selection sort using function.

30. Write a program to count the number of vowels in a string.

31. Write a program to concatenate two strings using function (without using library function).

32. Write a program to convert a string from upper case to lower case and vice versa.

33. Write a program to swap two numbers using function (pass the pointers).

34. Write a program to sort n number of strings in ascending order using pointer.

35. Write a program using pointers to copy a string to another string variable (without using library function).

36. Declare a structure of a student with details like roll number, student name and total marks. Using this, declare an array with 50 elements. Write a program to read details of n students and print the list of students who have scored 75 marks and above.

37. Create a structure to store the following information of employees.

a. Employee's number, name, pay and date of joining.

It has been decided to increase the pay as per the following rules:

Pay \leq Rs.3000 : 20% increase

Pay \leq Rs.6000 but $>$ Rs.3000 : 15% increase

Pay $>$ Rs.6000 : no increase

Write a program to implement the above structure.

38. Write a program to read a text file and count the number of vowels in the text file.

39. Write a program to copy a text file to another file.

2.5 LABORATORY

Covering paper 2.2

Each student should do at least 6 assignments (tasks) from the following list.

Objectives:

The Practical introduces the students to a personal computer and its basic peripherals, the process of assembling a personal computer, installation of system software like Windows OS, LINUX OS, device drivers. Basic system administration in Linux which includes: Basic Linux commands in bash, Create hard and symbolic links, Text processing, Using wildcards In addition hardware and software level troubleshooting process, tips and tricks would be covered.

Different ways of hooking the PC on to the network and internet from home and workplace and effectively usage of the internet. Configuring the TCP/IP setting. Usage of web browsers, email, newsgroups and discussion forums would be covered. In addition, awareness of cyber hygiene, i.e., protecting the personal computer from getting infected with the viruses, worms and other cyber attacks would be introduced.

Task 1: Identify the peripherals of a computer, components in a CPU and its functions. Draw the block diagram of the CPU along with the configuration of each peripheral and submit to your instructor.

Task 2: Every student should disassemble and assemble the PC back to working condition. Lab instructors should verify the work and follow it up with a Viva.

Task 3: Every student should individually install MS windows on the personal computer. Lab instructor should verify the installation and follow it up with a Viva.

Task 4: Every student should install Linux on the computer. This computer should have windows installed. The system should be configured as dual boot with both windows and Linux. Lab instructors should verify the installation and follow it up with a Viva

Task 5: Basic commands in Linux

Task 6: Hardware Troubleshooting: Students have to be given a PC which does not boot due to improper assembly or defective peripherals. They should identify the problem and fix it to get the computer back to working condition. The work done should be verified by the instructor and followed up with a Viva

Task 7: Software Troubleshooting: Students have to be given a malfunctioning CPU due to system software problems. They should identify the problem and fix it to get the computer back to working condition. The work done should be verified by the instructor and followed up with a Viva.

Task 8: The test consists of various systems with Hardware / Software related troubles, Formatted disks without operating systems. Installation of antivirus software, configure their personal firewall and windows update on their computer. Then they need to customize their browsers to block pop ups, block active x downloads to avoid viruses and/or worms.

Covering paper 2.3

Write programs using C language

Each student should do at least 15 assignments from the following list.

1. Implement binary search and linear search algorithms on arrays.
2. Implement following sorting algorithms :
 - i) Bubble sorting
 - ii) Insertion sort
 - iii) Heap sort
 - iv) Quick sort
 - v) Merge sort
3. Write a program to create a singly linked list and insert an element at the beginning, end, and at a given position of the linked list.
4. Write a program to create a singly linked list and delete an element from any position of the linked list.
5. Write a program to create a singly linked list. Write functions for
 - i. counting the number of elements in a list
 - ii. to search for a given element in a list. If the item has been found then it should return the position at which the item was found; otherwise it should return -1 to indicate not found.
6. Write a function to concatenate two linked lists.
7. Write a function to merge two sorted linked lists.
8. Write a program to create a doubly linked list and insert an element at any position.
9. Write a program to create a doubly linked list and delete an element from a given position.
10. Write a program to create a circular linked list and insert / delete an element at any position.
11. Write a program to implement a stack using
 - i) array structure
 - ii) linked list structure
12. Write a program to implement two stacks using a single array.

13. Write a program to evaluate a postfix expression using stack.
14. Write a program to convert an infix expression into a postfix expression.
15. Write a program to implement a queue using array.
16. Write a program to implement a queue using linked list.
17. Write a program to implement a circular queue using array.
18. Write a program to implement a circular queue using linked list.
19. Write a program to create a binary search tree using link representation and display the elements in preorder, in order and post order using recursive function.
20. Write a program to create a binary search tree using link representation and
 - i) search
 - ii) delete an item from the binary search tree.

Covering paper 2.4

Each student should do at least 12 assignments from the following list.

1. Write a program to create a child process that starts looping and then terminates.
2. Write a program to Show that the child can be set up to ignore a signal from its parent.
3. Write a program to Show that a process can ignore a signal.
4. Write a program to Create a thread in which prints "We are proud to be Indians" and terminates.
5. Write a program to demonstrate how to "wait" for thread completions by using the Pthread join routine. Threads are explicitly created in a joinable state.
6. Write a program to create a thread in which print "We are proud to be Indians" and pass multiple arguments using structure during its creation.
7. Write a program to compute the dot product of two vectors.
8. Write a program to compute the dot product of two vectors and also show the use of mutex variables.
9. Write a program to create threads, the main thread creates three threads. Two of these threads increment a counter variable while third thread watches the value of the counter variable. When the counter variable reaches a predefined limit, the waiting thread is signaled by one of the incrementing threads. The waiting thread "awakens" and then modifies the counter. The program continues until the incrementing threads reach a final value and also print the final value.
10. Write a program to fork() a child process so that we have two processes running: Each process communicates via a semaphore. The respective process can only do its work (not much here) When it notices that the semaphore track is free when it returns to 0. Each process must modify the semaphore accordingly.

11. Write a program to show how 2 processes can talk to each other using kill() and signal(). We will fork() 2 process and let the parent send a few signals to it's child.
12. Write a program to show attaching and detaching shared memory.
13. Write a program to show the communication between two processes through shared memory.
14. Write a program to implement Banker's Algorithm.
15. Write a program to simulate synchronization of Sleeping Barber problem.
16. Write a program to simulate Dining Philosophers Algorithm.

3.5 LABORATORY

Covering paper 3.3

Each student should do at least 12 assignments from the following list.

1. Define a class named *triangle* to represent a triangle using the lengths of the three sides. Write a constructor to initialize objects of this class, given the lengths of the sides. Write member functions to check
 - (a) if a triangle is isosceles
 - (b) if a triangle is equilateral.
 Write a main function to test your functions.

2. Define a structure **employee** with the following specifications.

Empno: Integer
Ename: 20 character
Basic, hra, da: float

Calculate() : a function to compute net pay as basic+hra+da with float return type.
Getdata(): a function to read values for empno, ename, basic, hra, da.
Dispdata(): a function to display all the data on the screen

Write a main program to test the program.

3. Define a class *circle* to represent circles. Add a data member *radius* to store the radius of a circle. Write member functions area() and perimeter() to compute the area and perimeter of a circle.
4. Define a class *complex* with two data members *real* and *imag* to represent real and imaginary part of a complex number.

Write member functions

rpart(): to return the real part of a complex number

ipart() : to return the imaginary part of a complex number

Add() : to add two complex numbers.

Mul() : to multiply two complex numbers.

Write constructors with zero, one and two arguments to initialize objects.

// This is an example of polymorphism.

5. Define a class *point* with two data members *xordinate* and *yordinate* to represent all points in the two dimensional plane by storing their x co-ordinate and y co-ordinate values.

Write member functions

dist(): to return the distance of the point from the origin.

slope(): to return the slope of the line obtained by joining this point with the origin.

Write constructors with zero, one and two arguments to initialize objects.

Write a friend function to compute the distance between two points.

6. Define a class *String* with the following data members

char *p;

int size

and write member functions to do the following (without using library function) and using dynamic memory allocation.

- Length of the string
- Compare two string
- Copy one string to another.
- Reverse the string.
- Write suitable constructors and destructors. Write a copy constructor for the class.

7. For the class *complex* defined in 3 above, overload the <<, >>, + and * operators in the usual sense. Also overload the unary – operator.
8. For the class *string* defined in 5 above, overload the <<, >> and + operators where + is to be used for concatenating two strings.
9. Define a class *time* to store time as hour, minute and second, all being integer values. Write member functions to display time in standard formats. Also overload the ++ and -- operators to increase and decrease a given time by one second where the minute and hour values will have to be updated whenever necessary.
10. Define a class to store matrices. Write suitable friend functions to add and multiply two matrices
11. Write a program to show the implementation of static members using class.

12. Define a class *student* with the following specification:

rollno

:integer

sname :20 characters

Derive two classes *artsst* and *scst*. The class *artsst* will represent students belonging to Arts stream and the class *scst* will represent students belonging to science stream. The *artsst* class will have additional data members *ph*, *hs*, *en* and *as* to store marks obtained by a student in three subjects Philosophy, History, English and Assamese. The class *scst* will have additional data members *ph*, *ch*, *ma* and *eg* to store marks obtained in Physics, Chemistry, Mathematics and English.

Write the following member functions in the classes *artsst* and *scst*

ctotal() : A function to calculate the total marks obtained by a student

takedata() : function to accept values of the data members

Showdata() : function to display the mark sheet of a student .

13. Define an abstract base class *printer*. Derive three classes, *laser-printer*, *dmp*, *line-printer* and *ink-jet-printer*. The derived classes will have data members to store the features of that particular printer. Write pure virtual function *display()* in the base class and redefine it in the derived classes.

14. Define a abstract base class *figure* and add to it pure virtual functions

display() to display a figure

get() to input parameters of the figure

area() to compute the area of a figure

perimeter() to compute the perimeter of a figure.

Derive three classes *circle*, *rectangle* and *triangle* from it. A circle is to be represented by its radius, rectangle by its length and breadth and triangle by the lengths of its sides. Write a main function and write necessary statements to achieve run time polymorphism.

15. Write an interactive program to compute square root of a number. The input value must be tested for validity. If it is negative, the user defined function *my_sqrt()* should raise an exception.
16. Define a class *rational* to store rational numbers as a pair of integers, representing the numerator and denominator. Write a member functions for setting the values of the numerator and denominator. This function should raise an exception if attempt is made to set a zero value as the denominator and in such cases it should be set to 1.
17. Write a class template for storing an array of elements. Overload the << and >> operators. Write a member function to sort the array in descending order.
18. Write a class template for representing a singly linked list. Write functions for inserting, deleting, searching and for displaying a linked list. Write a main function to test it on a linked list of integers and characters.

Each student should do at least 6 assignments from the following list.

1. Create a table *Employee* with the following columns:

- Emp_no (numeric) primary key
- Emp_name (string)
- Join_date (Date)
- Basic_pay_fixed_at (numeric)
- Date_of_birth (Date)

Insert the following data into the table.

Emp_no.	Emp_name	Join_date	Basic_pay_fixed_at	Date_of_birth
1001	Charles Babbage	01-Jun-2000	8000.00	03-10-1973
1002	George Boole	01-Jul-2001	5000.00	04-12-1972
1003	E.F. Codd	01-Jun-2001	8000.00	06-03-1969
1004	Bill Gates	01-Jul-2003	5000.00	09-10-1995
1005	Tony Greig	01-Aug-2004	8000.00	04-05-1985

2. Create the following two tables and insert data into the tables.

Player (Roll no.→Primary Key)

Roll no.	Name
10	Vijay Amrithraj
20	Leander Paes
30	Mahesh Bhupathi
40	Sania Mirza

Match (Match_no→Primary key, Roll no→Foreign key)

Match_no	Roll_no.	Match_Date	Opponent
1	20	10-Jul-2008	Washington
2	30	12-Jan-2008	Sampras
3	20	12-Aug-2008	Borg
4	30	20-Mar-2008	Vijay

Perform the following two operations:

- (i) Perform EQUIJOIN operation to retrieve data from both the files.

(ii) Perform OUTERJOIN operation to retrieve the unmatched records.

3. Design an ER diagram for a **BANK** database schema. To consider that each Bank can have multiple branches, and each branch can have multiple Accounts and Loans for customer. Also to specify the non weak & weak entity types, key attributes & key types, relationship types, instances, constraints and participations.

3. Create a table **Student** taking the attributes given bellow

Roll_no, Student_name, Address, Date_of_admission, Class
Section and Contact_no.

Write appropriate queries to perform the following operations:

- a) To insert values in the Student table.
- b) To delete values from Student table
- c) To list the names of all students which roll_no > 20.
- d) To search for students who got admitted before 01-01-2006.
- e) To change the name of the student whose roll number is 10 to Amar.

4. Create tables **Department** and **Employee** with the attributes given bellow.

Employee (EmpNo, Empname, Address, Dno)
Department (Dno, Dname, Location)

Dno in Employee is a foreign key.

Write appropriate queries to perform the following operations:

- a) To insert values in the tables.
- b) To retrieve the names and addresses of all Employees working in the Finance department.
- c) To print the location where Administration department is located.
- d) to delete all information regarding a particular employee.

5. Create table **Student** and **Course** taking the attributes given bellow.

Student (Roll_no, Name, Semester, Course_no(Foreign key))
Course (Course_no, Course_name)

Write appropriate queries for the following operations:

- a) To retrieve names of all students who are admitted to the course 'BCA'.

b) To retrieve the names of all students whose course name is 'BCA' and who are in the 3rd semester.

c) To display details of all courses in which more than 100 students got admitted.

d) For course being offered, display the course name and number of students admitted to the course.

6. Create tables *Employee*, *Department*, *Location*, *Works_on*, and *Project* taking the attributes given below.

Employee (Fname, Lname, Empno, Bdate, Address, Salary, Dnumber)

Department (Dname,Dnumber,Mgrno)

Locations(Dnumber, DLocation)

Works_on(Empno, Pnumber, Hours_per_day)

Project(Pname, Pnumber, Location,Dnumber (Foreign))

Dependent(Empno, Dependent_name, Sex, DOB, Relationship)

Write appropriate queries for the following operations:

a) Retrieve the names and addresses of all employees who work in the Finance department.

b) To retrieve the names of all employees who works on all the projects controlled by department number 6

c) For each department, print the name of the department and the name of the manager of the department.

d) Retrieve the location where the Administration department is located.

e) For every project located in Mumbai list the project number, the controlling department and department manger's name and address.

f) Find out how many employees are there in each department.

g) Find the total salary of all employees of the "Research" department, as well as the maximum, minimum and average salary in this department

h) Retrieve the name of all employees who have no dependent.

i) Alter the "Employee" table by deleting the coloumn Bdate.

j) Retrieve the Fname, Lname of all employees whose salary is higher than average salary.

k) For each department retrieve the department number, the number of employee in the department and their average salary.

l) Retrieve the name of all employees who have two or more dependent

m) Retrieve the details of all employees who works on project number 1,2,3

7. Create Table

Client_master (Client_no, name, address, Bdue)

Product_master(P_number,Description,saleprice,costprice)

Sales_master(Salesmno,Sname,Addres,Salamnt,Remarks)

Sales_order(O_no,Client_no,Odate,Delyaddr,Salesmno)

Sales_order_detail(Order_no,Product_no, Qtyorder, product_rate,Qty_dispatched)

Write appropriate queries to perform the following operations:

- i) List name of all clients having 'a' as the second letter in their names.
- ii) Retrieve the description and total Qty sold for each product.
- iii) Find product no. and description of non moving products (i.e product not being sold).
- iv) For each product being sold, list the product number and the total amount (in Rs.) sold.
- v) List all client who stay in 'Bangalore' or 'Mumbai'
- vi) List the clients who stay in a city whose First letter is 'M'
- vii) Find the names of clients who had purchased the item 'Trouser'.
- viii) Find out if 'T-Shirt' has been ordered by any client and if so print the details of the client.
- ix) List details of all products whose unit price is more than Rs. 5000.00.
- x) Calculate the total amount (in Rs.) purchased by each client that has purchased items amounting more than Rs. 20000.

8. Create table

Author(Author_id, Name, City, Country)

Catalog (Book_id, Title, Author1_id, Author2_id, Publisher_id, Category_id, Year, Price)

Publisher(Publisher_id, Name, City, Country)

Order_details(Order_no, Book_id, Quantity)

Category(Category_id, Description)

Order_summary(Order_no,Member_id,Odate,Amount,Ostatus)

Member(Member_id, Name, Address, Contact)

Assume that all books have at most two authors.

Write appropriate queries to perform the following operations:

- a) Retrieve the title, author, and publisher names of all books published in 1999 and 2006.
- b) Retrieve the title of all books whose one author is 'A Tanenbum'.
- c) Get the details of all books whose price is greater than the average price of the books.
- d) Get the names of all the books for which an order has been placed.
- e) Get the names of all authors who have more than ten books in the catalog.
- f) Get the details of the authors whose books are being sold from the book club.
- g) Get the title and price of all books whose price is greater than the maximum of the category average.

4.5 LABORATORY

Covering paper 4.1

Each student should do at least 8 assignments from the following list.

1. Design a class to represent a bank account and include the following data members –
Data Members: name of the depositor, account number, type of a/c, balance amount in the a/c
Methods: to assign initial values, to deposit an amount, to withdraw an amount after checking the minimum balance (Rs.1000), to display the name of the depositor and balance.
2. Write an applet programming to print the first name, last name, sex, address, mobile no. and pin code of an end user passing parameters.
3. Write an applet programming to create three buttons and draw a rectangle on clicking the first button, a solid rounded rectangle on clicking the second button and a solid circle and an arc on clicking the third button.
4. Write a program to create 3 – threads for execution with different priorities.
5. Write a program to create three threads for execution of the natural nos. less than 5 using synchronization concept.
6. Write a java program for a class teacher that contains two fields name and qualification. Extend the class to department that contains data members deptno and deptname. An interface name as college contains one field name of the college. Using the above classes and interface get the appropriate information and display them.
7. Design three classes person, employee and student using the concept of inheritance. Each class should have a constructor of its own properties as name, age, gender and common method showdata().

8. Write a program to create an array of employee name and salary related to the employee. If the salary is greater than Rs.10,000 raise an exception “Salary is greater than Rs.10,000”, otherwise display the required information.
9. Design a user interface to insert, update, delete, search and browse a record of employee database for emp_id, emp_name, sex, basic, desig and date_join.
10. Design a menu as Operation and Exit. The Operation menu contains the menu items as Addition, Subtraction, Multiplication, Division and Remainder of two numbers. On clicking the respective menu items perform the desired operation. Further on clicking the Exit menu put a confirmation message as “Do you want to exit- Yes/No?”. If Yes then terminate the application, otherwise it exists.

Covering paper 4.2

Each student should do at least 3 assignments from the following list.

1. Xpaize.com wants to computerize its sales, purchase & booking of shares through online mode .
A user enters his/her user name and password (assigned at the time of registration).

User chooses one of the above actions and finally receives the trasaction/account details on the action chosen.

Perform the following tasks for the above problem.

- a) Develop SRS.
- b) Draw DFDs of level 0 and level 1.
- c) Draw an ER diagram and its related tables.

2. The university wants to computerize its admission process. The system should maintain data of all new students as well as old students , their results issue admission letter , allocate study centre or should also allow students to move from one centre to another study centre.

Perform the following tasks for the above problem.

- a) Develop SRS.
- b) Draw DFDs of level 0 and level 1.
- c) Draw an ER diagram and its related tables.

3. A Tic – Tac - Toe is a computer game in which human player and computer makes alternative moves on a 3X3 square. A move consists of marking a previously unmarked square. The player who is first to place three consecutive marks along a straight line (ie along a row , column or diagonal) on the square wins . As soon as the either of the winner wins a message congratulating them is displayed .If neither of the player manages to get three consecutive marks along a straight line and all the squares on the board are filled up , then the game is drawn. The computer always tries to win a game.

Perform the following tasks for the above problem.

- a) Draw DFDs of level 0 and level 1.
- b) Draw an ER diagram and its related tables.

4. A supermarket needs to develop the following software to encourage regular customers. For this the customer needs to supply his /her residence address ,telephone number & the driving licence number.Each customer who is registered for this scheme is assigned a unique customer number (CN)by the computer. Based on the generated CN a clerk manually prepares a customer identity card after getting market managers signature on it . A customer can present his customer identity card to the check out staff whenever he makes any purchases.In this case the value of his purchase is credited against his CN.At the end of each year the supermarket intends to award surprise gifts to ten customers who make the highest total purchase over the year.Alsp it intends to award a 22 caret gold coin to every customer whose purchase exceeded Rs. 10,000. The entries against the CN are reset on the last day of every year after the prize winner list are generated.

Perform the following tasks for the above problem.

- c) Draw DFDs of level 0 and level 1.
- d) Draw an ER diagram and its related tables.

Covering paper 4.3

Each student should do at least 4 assignments from the following list.

Avoid using Loopback Communication for the assignments,1-4:

1. Write a server socket program using TCP/IP in java where the client side will send a request for an existing file to the server side and if the file exists in the server then send the contents of that particular file to the client in reply, otherwise display a message “file does not exist on the server”, if exists displays the contents on the client side.
2. Develop a chat application using TCP/IP in java.
3. Develop a client-server application using TCP/IP in java to input user’s information and finally send them to the server and store there in a file.

4. Develop a server socket program where the client takes principal, rate of interest and number of years and send them to the server. In the server receive this information and find the simple interest and finally send the result to the client again and display it on the client's VDU.
5. Configure a Local Area Network (Wired/Ethernet) in Linux environment. Configure the network interface card using ifconfig command and also explore ping, ifdown and ifup commands.

5.5 LABORATORY

Covering paper 5.2

HTML

(At least 12 assignments has to be done from this group)

1. Create a HTML document consisting of HTML heading, paragraphs and images.
2. Create a HTML document and insert comments in the HTML source code and insert horizontal lines.
3. Construct HTML document to set the font of a text , size of the font, style of the font.
4. Create a HTML document to show how to create hyperlinks.
5. Create a HTML document to use an image as a link.
6. Create a HTML document to open link in a new browser window.
7. Create a HTML document to jump to another part of a document (on the same page).
8. Create a HTML document to insert images from another folder or another server.
9. Create an image-map, with clickable regions.

10. Create a HTML document with all table elements (Table, Caption, Table Row, Table Data element, Table Heading Element, THEAD, TFOOT, TBODY)
11. Create HTML document to make an unordered list, an ordered list, different types of ordered lists, different types of unordered lists, Nested list, Definition list.
12. Create HTML form with the all FORM elements (text fields, password field, Checkboxes, Radio buttons, Select elements, Drop-down list with a pre-selected value, Textarea (a multi-line text input field) and buttons.
13. Create HTML document with all Frame elements (FRAMESET, FRAME, NOFRAMES, and INLINE FRAME).
14. Create a HTML document to add AUDIO and VIDEO.
15. Create a HTML document to aligning images
(Let the image float to the left/right of a paragraph)
16. Create a HTML document to jump to a specified section within a frame
17. Construct a HTML document with CSS to Set the background colour of a page.
18. Construct a HTML document with CSS to set an image as the background of a page
19. Construct HTML document with CSS to Set the text color of different elements and align the text.
20. Construct HTML document to set different colours to visited/unvisited links, Specify a background colour for links

XML

Each student should do at least 2 assignments from the following list

21. Construct an XML document that contain information about products of an organization.
22. Construct an XML document that contain information of 5 students (such as roll no., name, address, class).
23. Construct an XML document that contain details of 10 books.

JAVAScript

(At least 8 assignments has to be done from this group)

24. Write a program in javascript to accept a name from the user and display the same name in an alert box.
25. Write a program in javascript to display a message in a confirm box.
26. Write a program in javascript to display the message 'time is running out' in the status bar.
27. Write a program in JavaScript to enter marks of a student and find his/her grade according to the following:
if marks >= 90 then grade A
if marks >= 80 then grade B
if marks >= 70 then grade C
if marks >= 60 then grade D
else fail.
28. Write a program in JavaScript to create a button and when the button is clicked the message 'Hello World' is displayed on an alert box..
29. Write a program in JavaScript to accept 2 nos from the user and show the working of all arithmetic operators.
30. Write a program in JavaScript to accept 2 strings and concatenate them.
31. Write a program in JavaScript to display the current date and time.
32. Write a program in JavaScript to find the length of an array.
33. Write a program in JavaScript to check whether a string is palindrome or not.
34. Write a program in JavaScript that responds to a mouse click anywhere on the page
(using mouse click).
35. Write a program in JavaScript to display the contents of a check box in a alert box.
36. Write a program to validate a form in the user id and password forms.
37. Write a program in JavaScript to create a welcome cookie, Button animation, Image map with added JavaScript
Simple timing, Timing event in an infinite loop

VBScript

(At least 4 assignments has to be done from this group)

38. Write a program in VBScript to create a variable.
39. Write a program in VBScript to uppercase to lowercase.
40. Write a program in VBScript to Create an array

41. Write a program in VBScript using conditional statements loop
42. Write a program in VBScript using loop.
43. Write a program in VBScript to display Date and Time
44. Write a program in VBScript to display the current month and day.

ASP

Each student should do at least 2 assignments from the following list

45. Write a program in ASP to interact with a user in a form that uses the "get" method.
46. Write a program in ASP to interact with a user in a form that uses the "post" method.
47. Write a program in ASP to interact with a user in a form with radio buttons.
48. Write a program in ASP to create a welcome cookie.

Covering paper 5.4

Each student should do at least 8 assignments from the following list

1. Write a program to add two 8 bit numbers & store it in a memory location 8820h.
2. Write a program to copy a block of memory from one location 8820h to another location 8840h.
3. Write a program to perform the addition of two 16 bit numbers.
4. Write a program to add two numbers & store it in a register e.
5. Write a program to load two unsigned numbers in register b & c. Subtract b from c. If the result is in 2's complement, convert the result in absolute magnitude & display it.
6. Write a program to find the difference of two numbers & store the result in a memory location 8830h.
7. Write a program to find the larger / smaller of two given numbers.
8. Write a program to subtract two numbers and add it to a given memory location.
9. Write a program to perform $x+y-5$, where x and y are 16-bit numbers.
10. Write a program to find 2's complement of a number.

6.5 LABORATORY

Each student should do at least 15 assignments from the following list.

1. Installation of Linux Operating System and partitioning the disk.
2. Installing software packages in linux OS using GUI as well as command line.
3. Changing the default run level of a system
4. Mounting and un mounting a removable media.
5. Finding the list of all running processes and redirect the output in a file.
6. Use of different kill signals to kill a running process.
7. Bringing a process from back ground to fore ground and vice-versa.
8. Adding and managing user accounts.
9. Monitoring disk space quota and memory usage and redirect the output in a file.
10. Backup and restoring a file.
11. Compression and extracting a file. Use command line.
12. Configuring a network interface and assigning a default route.
13. Scheduling job using crontab.
14. Changing the ownership and access permission of file or directory. Use command line.
15. Copy, move and rename a file.
16. Configuring a ftp server
17. Assigning address of DNS.
18. Use of ssh, telnet, netstat, ping, route commands.
19. Use grep, awk, sed commands.
20. Use of redirection and piping.
21. Monitoring and managing system log information.
22. Basics of firewall using iptables.
23. Basics of configuring http server.
24. Managing different services in linux.
25. Monitoring the traffic going through a network interface.
26. Write shell script to
 - a. Find factorial of a given number
 - b. Convert a decimal number to hexadecimal number