

SYLLABUS FOR 3 YEAR B.Sc. (HONOURS) COURSE
IN
COMPUTER SCIENCE

PART – I

PAPER – I

(THEORETICAL: FULL MARKS 100)

1.1	Computer Fundamentals		(30)
1.1.1	Introduction to Computers & Problem Solving	(10)	
1.1.2	Number Systems & Arithmetic	(10)	
1.1.3	Boolean Algebra & Digital Logic	(10)	
1.2	Digital Logic Design & Computer Organization – I		(70)
1.2.1	Introduction to Basic Electronics	(10)	
1.2.2	Digital Logic & System Design	(30)	
1.2.3	Computer Organization – I	(30)	

PAPER – II

(THEORETICAL: FULL MARKS 100)

2.1	Graph Theory		(25)
2.2	Discrete Mathematics		(25)
2.3	Numerical Algorithms	(30)	
2.4	Operation Research		(20)

PART – II

PAPER – III

(THEORETICAL: FULL MARKS 100)

3.1	Programming Language – I		(25)
3.1.1	Introduction to Programming Paradigms & Techniques	(10)	
3.1.2	C Programming	(15)	
3.2	Data Structures		(25)
3.3	Systems Software		(50)
3.3.1	Operating System	(25)	
3.3.2	Formal Language & Language Translators	(25)	

PAPER—IV

(PRACTICAL: FULL MARKS 100)

GROUP — A (SOFTWARE LABORATORY — I: MARKS 50)

- | | | |
|-----|--|------|
| 4.1 | Familiarity with Single & Multi user Operating Systems | (10) |
| 4.2 | Familiarity with Window Based PC Software | (10) |
| 4.3 | Programming with C Language | (30) |

GROUP — B (HARDWARE LABORATORY — I: MARKS 50)

- | | | |
|-----|----------------------|------|
| 4.4 | Digital Logic Design | (50) |
|-----|----------------------|------|

PART - III

PAPER — V (P51)

(THEORETICAL: FULL MARKS 100)

- | | | | |
|-----|--|--------|------|
| 5.1 | Microprocessor & Computer Organization — II | | (50) |
| | 5.1.1 Microprocessor, Interfacing & its Applications | (25) ✓ | |
| | 5.1.2 Computer Organization — II | (25) | |
| 5.2 | Data Communication & Computer Networks | | (30) |
| | 5.2.1 Data Communication | (10) | |
| | 5.2.2 Computer Networks | (20) | |
| 5.3 | Computer Graphics | | (20) |

PAPER—VI (W)

(THEORETICAL: FULL MARKS 100)

- | | | | |
|-----|--|------|------|
| 6.1 | Programming Language —II | | (40) |
| | 6.1.1 Object Oriented Programming Concepts | (15) | |
| | 6.1.2 Visual Programming | (10) | |
| | 6.1.3 Internet Technologies | (15) | |
| 6.2 | Software Engineering | | (20) |
| 6.3 | Data Base Management System | | (40) |

PAPER-VII

(PRACTICAL: FULL MARKS 100)

GROUP - A (HARDWARE LABORATORY: MARKS 50)

7.1 Microprocessor Programming & I/O Interfacing (P.S.)

GROUP - B (SOFTWARE LABORATORY: MARKS 50)

7.2 System Configuration (10)

7.3 Front End Programming & WEB Utilities [VB, HTML] (14) (40)

PAPER - VIII

(PRACTICAL: FULL MARKS 100)

8.1 C / C++ Programming (P.S.) (50)

8.2 Relational DataBase (ORACLE / SQL SERVER/ SYBES) (14) (30)

8.3 Unix Shell Programming (14) (20)

PART - I

Paper - I

1.1 Computer Fundamentals

1.1.1 Introduction to Computers & Problem solving

Generation of Computers; Computer system: Basic Block Diagram, Super Mainframe, Mini & Personal Computer, Nomenclature, **Software:** Systems and Application; **Hardware & Software; Algorithms:** Definition, essential features; **Complexity:** notation, time & space; **Computability & correctness concepts; Structured programming concepts; Process of problem solving, Flowcharts and Pseudo codes.**

1.1.2 Number System & Arithmetic

Number System: Positional, binary, octal, decimal, hexadecimal and their representations, **Methods of conversion from one base to another;** sign magnitude, 1's complement; 2's complement; **Binary Arithmetic; Fixed & floating point numbers:** representation, biased exponent, range & precision, errors, overflow, underflow, BCD arithmetic

1.1.3 Boolean Algebra

Concepts of propositional logic; Boolean algebra: definitions, postulates, properties, simplification of logical expressions using properties and maps (up to 4 variables), Min-term, Max-term expressions; **Logic gates: AND OR, NOT, XOR, Combinational circuits, Simple** logic design using logic gates.

1.2 Digital logic design & Computer Organization - I

1.2.1 Introduction to Basic electronics

Elementary physics of semiconductor: P-N junction diodes, Zener diodes, FET, FET, MOSFET; **Equivalent circuits for diodes, transistors, FETS;** **Switching characteristics of diodes and transistor, Thyristor;** **Applications of diodes in rectification and clamping;** **Amplifiers:** Class A, B, C, Multistage amplifiers; **Concept of negative & positive feedback;** **Oscillators: R-C, Crystal;** **Regulated Power Supply :** shunt & series; switching; **Inverters:** using bi-polar transistors & FET; **Operational Amplifiers.**

1.2.2 Digital Logic and System Design

Alphanumeric codes: ASCII, EBCDIC; **Single error detection & correction; Hamming Codes and decoding techniques.** **Combinational Circuits:** encoder, decoder, code converter, comparator, adder/subtractor, multiplexer, demultiplexer, parity generator, PLA. **Basic Sequential circuits; Flip-Flops: RS, JK, D, T; Schmitt-trigger; Concepts of single and two-phase clocks; Multivibrators:** astable and monostable; **Registers and shift registers; Counters:** Synchronous & asynchronous. **Digital-Analog interfacing:** D/A & A/D converters, basic ladder, counter ramp, successive approximation, sample and hold.

1.2.3 Computer Organization — I

Computers: Basic building blocks and subsystems, Control and intra-connection, **Bus; Tri-state devices; CPU:** General structure, **CPU Registers; Instruction formats & Design strategies, Addressing modes:** direct, indirect, immediate, relative, implied, indexed, 0-1-2 addressing schemes; **Stack organization & implementation; Memory:** Organization, addressing, reading & writing, SAM, RAM & ROM, different storage technology.

Paper—II

2.1 Graph Theory

Graphs: Definition, finite and infinite graphs, directed and undirected, incidence, degree, isolated vertex, pendant vertex; null graphs. **Walks; paths and circuits; Connected and disconnected graphs; Euler's graph; Hamiltonian paths and circuits** **Trees:** Definition and basic properties; distance and centers; spanning trees; [Shortest path problems;] **Matrix representation of graphs:** Incidence, adjacency and circuit matrix, simple applications.

2.2 Discrete Mathematics

Permutation; Combination; Groups; Subgroups; groups within groups; normal subgroups; Rings; Sub rings; Fields; Vector spaces.

2.3 Numerical Algorithms

Concept of errors in computation; Solution of non-linear equations: Initial value and convergence, Bisection, Regula-Falsi, Newton-Raphson; **Solution of linear equations of the form $Ax=b$:** Gaussian elimination, Matrix inversion, triangular factorization, Gauss-Seidel; **Eigenvalues and Eigenvectors:** definitions, properties, **Interpolation:** Newton's forward and backward interpolation, Lagrange interpolation; **Least square method of curve fitting;**

Numerical integration: - Trapezoidal rule, Simpson's rule, Gauss quadrature formula;
Solution of differential equations: Euler's method, Taylor's series, Runge-Kutta method;

2.4 Operation Research

Review of linear programming; (Transportation and assignment problem)
Integer programming: Cutting plane, branch and bound; **Dynamic programming;**
Modeling; Optimization; PERT/CPM networks; Queuing theory: Basic concepts, queuing models, Poisson statistics, M/M/I queue; **Applications.**

PART - II

✓ Paper—III

3.1 Programming Language — I

3.1.1 Introduction to Programming Paradigms & Techniques

Criteria for Language design; Character sets; BNF; Variables & Assignment statements; Constants & initialization; Conditional & iterative statements; Data Types: Elementary, Enumerated, Pointer, Structure, type collection, **Scope & Extents, Parameter Evaluation and Passing:** Call by name, call by value, call by reference; **Data abstraction.**

3.1.2 ✓ C Programming

Introduction: Basic structure, character sets, keywords, identifiers, constants, variable type declaration, **Execution of some simple sample programs. Operators:** arithmetic, relational, logical and assignment, increment, decrement and conditional; **Operator precedence and associations; Expressions in C; Expression evaluation and type conversion; Formatted input & output; Statements in C:** Assignment, control and loops; **Arrays:** Single and multi dimensional, initialization, string handling with arrays, string handling functions. **Functions in C :** need, sample examples, argument passing in C, functions & their use, return values and their types, recursion; **Structures :** definitions & initialization, array of structure, array within structures. **Pointers:** declaration & initialization, accessing variables through pointers, pointer arithmetic, pointers & arrays; strings; pointer to functions & structures. **File access:** opening LK: closing, I/O operations; **Linked list:** Concepts, simple implementations.

3.2 ✓ Data Structures

Definitions : Concepts of data types, elementary structures, words and their interpretation; **Arrays :** Types, memory representation, address translation, functions of single and multi-dimensional arrays with examples; **Linked structures :** Singly and doubly linked list(non-circular and circular); **List manipulation with pointers :** Examples involving insertion and deletion of elements, **Stacks and queues :** Definition, representation, uses and applications Recursion, postfix conversion and evaluation, application of queues, **Binary trees** Definition, quantitative properties; **Path length :** internal and external, properties, minimum and- maximum path length of a binary tree, importance; **Searching :** Linear and binary- search, performance and complexity; **Hashing :** Concept, advantages and disadvantages, different types of hash function, collision resolution techniques-open addressing with probing, linear

coalesced chaining, applications; **Sorting:** terminology, performance evaluation Different sorting techniques (Bubble, insertion, selection, heap, partition exchange, radix) with

interactive and / or recursive description, complexity, advantages and disadvantages.

3.3. Systems Software

3.3.1 Operating Systems

✓ **What is OS; Concepts of processes; Files; shell; system calls; Structures:** Monolithic, layered, virtual, client-server and distributed model: **Concept of Synchronization:** Semaphores, critical regions, monitor etc., inter-process communication mechanisms; **Processor Management; Scheduling :** round-robin, priority, queue; **I/O Management :** Device and device controllers; interrupt handlers and device drivers; **Memory Management :** Multiprogramming, swapping, paging, virtual memory, page replacement techniques; **File system :** files and directories, file servers, security and protection, **Deadlock :** Definition, detection and prevention. **Case study:** DOS, UNIX, WINDOWS..

3.3.2 Formal Language & Language Translators

✓ **Introduction to formal languages and grammars; Finite automata:** Regular expressions, Deterministic and non-deterministic finite automata and their equivalence, state minimization, **General grammars; Turing machine; Language Translators:** Assemblers, Loaders, Linkers, Interpreter; **Compiler:** Various phases of compilation, cross compilers.

PART - III

Paper - V

5.1 Microprocessor and Computer Organization - II

5.1.1 Microprocessor, Interfacing and its applications

Evolution of microprocessors, Basic structure and programming, Clock cycles; timing diagrams; Interrupts; Bus standards and Interfacing concepts : Memory interfacing, I/O interfacing and Oils - keyboard interfacing, display interfacing, storage device interfacing, printer, **Programming a microprocessor :** Addressing, Data movement, Arithmetic and logical instruction; Interrupt handling; **Assemblers, interrupt:** Methods of interrupt, priority & management.

5.1.2 Computer Organization - II

ALU : Main sub-units, registers, arithmetic operations, shift and data transfer operation storing, integer and floating point arithmetic circuits, **Memory Hierarchy :** CPU Registers, Cache Memory, Primary memory, Secondary Memory, Virtual Memory, **Control Unit :** Principles, sequence of operations, fetch-decode-execute cycle, micro instruction control organization, **Input-output (I/O):** Modes of data transfer, buffers, I/O channels, bus other interrupt processing levels and priorities; **Keyboard; VDU; Dot matrix printer and other devices.**

5.2 Data Communication & Computer Network

5.2.1 Data Communication

Data Communication; Transmission media; Network; Protocol and standard Analog & Digital signals; Periodic & non-periodic signals; Time & Frequency domains; Multiplexing: FDM, TDM and applications; **Encoding:** D/A, V/D encoding; Error: Different type of errors and their detection.

5.2.2 Computer Networks

Concepts of centralized and distributed computing: Advantages of networking; **Layered architecture, OSI architecture:** Basic features, LAN, MAN and WAN; **Simple P based network examples:** block diagram, mode of operation and characteristic features.

5.3 Computer Graphics

Introduction: Computer art, animation, image processing, morphing

Projection & Clipping

2-D & 3-D transformation

Liner, Curves and their representation

Paper — VI

6.1 Programming Language — II

6.1.1 Object Oriented Programming Concept,

Concepts; difference with procedure oriented programming; **Data abstraction** Objects, classes and methods, inheritance, polymorphism. **Software design:** problem with top down design; Object Oriented approach; **Structured analysis tools; Hybrid design, Principle of separation & responsibility based design.**

6.1.2 Visual Programming

Basic features: Building objects with classes; Operations with objects; Class libraries; Multitasking and multithreading; Forms; Objects; Events Functions; Procedures: Methods: ODBC Driver; Front end development with database; MFC.

6.1.3 Internet Technologies

What is Internet; Servers; Clients; Port; Domain Name Server (DNS); Accounts ISP; Connection: Dial Up, ISDN, ADSN, Cable modem; **Email:** Account, sending receiving, on mailing list, IRC; **Voice & Video conferencing WWW; Browser.**

6.2 Software Engineering

Software life cycle; Different Models; Software Definition; Software Requirement; Software Specification; Software Quality; Software Maintenance; Software Metric.

6.3 Data Base Management System

Basic concepts: Advantages of DBMS, ANSI/SPARC architecture, physical, conceptual and external models; **Entity Relationship diagrams; Data models:** Relational, hierarchical and network. **File organization:** Sequential, indexed sequential, random, inverted; **Query Languages:** Relational Algebra and Relational Calculus; **Fundamental dependencies and normal forms:** 1NF, 2NF, 3NF and BCNF, **Structured Query Language:** Elementary ideas and simple examples; **Security; Integrity; Case Study:** Any RDBMS.