Scheme of Programme-B.Sc Computer Science

(Scheme UG A1: Undergraduate Programmes(Multidisciplinary)

Semester 1

Course Code	Course Title	Course ID	L (Hrs)	T (Hrs)	P (Hrs)	L Credits	T Credits	P Credits	Total Credits	MARKS TI	MARKS TE	MARKS PI	MARKS PE	MARKS Total
Core Course(s)														
CC-A1	Computer Fundamental and Architecture	240/CS/CC101	3	0	2	3	0	1	4	25	50	5	20	100

Semester 2

Course Code	Course Title	Course ID	L (Hrs)	T (Hrs)	P (Hrs)	L Credits	T Credits	P Credits	Credits	MARKS TI	MARKS TE	MARKS PI	MARKS PE	MARKS Total
Core Course(s)														
CC-A2	Programming In C	240/CS/CC201	3	0	2	3	0	1	4	25	50	5	20	100

Semester 3

Course Code	Course Title	Course ID	L (Hrs)	T (Hrs)	P (Hrs)	L Credits	T Credits	P Credits	Credits	MARKS TI	MARKS TE	MARKS PI	MARKS PE	MARKS Total
Core Course(s)														
CC-A3	Computer Networks	240/CS/CC301	3	0	2	3	0	1	4	25	50	5	20	100

Semester 4

Course Code	Course Title	Course ID	L (Hrs)	T (Hrs)	P (Hrs)	L Credits	T Credits	P Credits	Credits	MARKS TI	MARKS TE	MARKS PI	MARKS PE	MARKS Total
Core Course(s)														
CC-A4	Data Structure with C/C++	240/CS/CC401	3	0	2	3	0	1	4	25	50	5	20	100

Semester 5

Course Code	Course Title	Course ID	L (Hrs)	T (Hrs)	P (Hrs)	L Credits	T Credits	P Credits	Credits	MARKS TI	MARKS TE	MARKS PI	MARKS PE	MARKS Total
Core Course(s)														
CC-A5	Operating System	240/CS/CC501	3	0	2	3	0	1	4	25	50	5	20	100

Internship is to be done during summer break after 4thSemester, Marks will be added in 5th Semester.

Semester 6

Course Code	Course Title	Course ID	L (Hrs)	T (Hrs)	P (Hrs)	L Credits	T Credits	P Credits	Credits	MARKS TI	MARKS TE	MARKS PI	MARKS PE	MARKS Total
Core Course(s)														
CC-A6	Software Engineering	240/CS/CC601	3	0	2	3	0	1	4	25	50	5	20	100

The curriculum of semester 7 and 8 will is provided in due course of time.

SEMESTER 1

CC-A1: COMPUTER FUNDAMENTAL AND ARCHITECTURE

Course code	CC-A1			
Category	Core Course			
Course title	Computer Fundamental and Architecture			
Course ID	240/CS/CC101			
Scheme and Credits	L	T	P	Credits
	3	0	2	4
Theory Internal	25 marks			
Theory External	50 marks			
Practical Internal	5 marks			
Practical External	20 marks			
Total	100 marks			
Duration of Exam	3 hrs			

COURSE OBJECTIVE: The aim of the course is to provide knowledge of computer as a system and making student aware of internal mechanism of computer hardware and its working.

UNIT – I

Number Systems: Binary, Octal, Decimal and Hexadecimal, Conversions from one number system to another, BCD Codes, Error Detecting and Correcting Codes, Character Representation – ASCII, EBCDIC and Unicode, Binary Arithmetic; Binary Addition, Binary Subtraction, Binary Multiplication, Binary Division, Complementary numbering systems: 1's and 2's Complements representations, Fixed-Point and Floating-Point Representation of Numbers.

UNIT – II

Boolean Algebra: Boolean Algebra Postulates, basic Boolean Theorems, Boolean Expressions, Boolean Functions, Truth Tables, Canonical Representation of Boolean Expressions: SOP and POS, Simplification of Boolean Expressions using Boolean Postulates & Theorems, Kaurnaugh-Maps (upto four variables), Tabular Method, Handling Don't Care conditions.

UNIT – III

Logic Gates: Basic Logic Gates – AND, OR, NOT, Universal Gates – NAND, NOR, Other Gates – XOR, XNOR etc. NAND, NOR. Their symbols, truth tables and Boolean expressions. Combinational Circuits: Design Procedures, 12 Half Adder, Full Adder, Half Subtractor, Full Subtracor, Multiplexers, Demultiplexers, Decoder, Encoder, Comparators, Code Converters.

UNIT – IV

Sequential Circuits: Basic Flip- Flops and their working. Synchronous and Asynchronous Flip –Flops, Triggering of Flip-Flops, Clocked RS, D Type, JK, T type and Master-Slave Flip-Flops. State Table, State Diagram and State Equations. Flip-flops characteristics & Excitation Tables. Designing Registers & Counters: Asynchronous and Synchronous Binary Ripple Counter, Binary Synchronous Counter, Modulo-N Counters and Up-Down Counters.

Text Books:

[1] M. Morris Mano, Digital Logic and Computer Design, Prentice Hall of India Pvt. Ltd.

[2] V. Rajaraman, T. Radhakrishnan, An Introduction to Digital Computer Design, Prentice Hall.

Reference Books:

[1] Andrew S. Tanenbaum, Structured Computer Organization, Prentice Hall of India Pvt. Ltd.

[2] Nicholas Carter, Schaum's Outlines Computer Architecture, Tata McGraw-Hill.

Practical List

1. Convert numbers between Binary, Octal, Decimal, and Hexadecimal using a program.
2. Implement the conversion between Binary Coded Decimal (BCD) and Binary systems, and vice versa, using a program.
3. Perform Binary Addition, Binary Subtraction, Binary Multiplication, and Binary Division. Implement these operations in a program.
4. Solve arithmetic problems using 1's and 2's complement representations for signed numbers. Implement a program that computes arithmetic operations using these complements.
5. Verify Boolean postulates and basic Boolean theorems (e.g., De Morgan's laws) using truth tables and manual derivation.
6. Express Boolean functions in Sum of Products (SOP) and Product of Sums (POS) canonical forms and converting between these forms manually.
7. Minimize Boolean expressions using K-Maps (up to 4 variables).
8. Design and implement basic logic gates (AND, OR, NOT) and universal gates (NAND, NOR). Show truth tables, Boolean expressions, and actual circuit designs.
9. Implement XOR and XNOR gates, show their truth tables and Boolean expressions, and demonstrate their use in circuits.
10. Design Half-Adder and Full-Adder circuits using logic gates. Create truth tables and verify the outputs.
11. Design Half-Subtractor and Full-Subtractor circuits using logic gates. Verify the truth tables and the correct subtraction output

Semester 2

PROGRAMMING IN C

Course code	**CC-A2**			
Category	Core Subject			
Course title	**Programming In C**			
Course ID	240/CS/CC201			
Scheme and Credits	L	T	P	Credits
	3	0	2	4
Theory Internal	25 marks			
Theory External	50 marks			
Practical Internal	5 marks			
Practical External	20 marks			
Total	100 marks			
Duration of Exam	3 hrs			

COURSE OBJECTIVE: Master foundational programming structures, data types, control flow mechanisms, and advanced concepts such as arrays, strings, functions, structures, unions, pointers, and file handling for effective problem-solving in C programming.

UNIT- I

Programming Structure: Sequence, Selection, Iteration and Modular. Problem Solving techniques: Development Tools: Algorithm, Flowcharts and Pseudo code (Definition and its characteristics) Developing Algorithm and Drawing flowcharts

UNIT- II

C Character set, Tokens, Identifier, Keywords, Variables, Data types, Qualifiers. Operators and Expressions: Arithmetic, Relational, Logical, Bit-Wise, Increment, Decrement, Conditional and Special operators. typedef, Type Conversion, Constants, Declaring Symbolic Constants, Character Strings, Enumerated Data Types, Operator Precedence and Associativity. Library functions. : Maths, string handling Functions. Control Structure: Compound Statement, Selection Statement: if, if-else, Nested if, switch. Iteration statement: for, while, do..while, Nested loops, Jump statement: break, continue, goto. (Special emphasis on problem solving)

UNIT- III

Arrays: Need, Types: Single and Two Dimensional Array. Strings: Strings Manipulation, Arrays of Strings, Evaluation order Function: Function Components, Return Data type, Parameter Passing, Return by Reference, Default Arguments, Recursive Functions, Arrays with Functions, Storage Classes.

UNIT- IV

Structure: Declaration, Definition, Accessing structure members, Initialization, Nesting of Structures. Union: Unions, Differences between Structure and Union Pointer: Introduction, Address Operator (&), Pointer variables, void pointers, Pointer Arithmetic, Pointers to Pointers. File handling : Hierarchy of File Stream Classes, Opening & closing a file, Testing for errors, File Modes, File pointers and their manipulations, Sequential Access, Random Access , Command Line arguments.

Text and Reference Books:

1. The Art of programming through flowcharts & algorithm by Anil B. Chaudhari Firewall Media, Laxmi publication, New Publication.
2. Programming in C by E. Balagurusamy TMH Publications.
3. C Programming - Kernighen Ritche
4. Programming with C – Y. Kanetkar.
5. C Programming – Holzner, PHI Publication.
6. Programming in C – Ravichandran.

List of Practical:

1. Write a program to display "Hello, World!" on the screen.
2. Write a program to perform addition, subtraction, multiplication, and division of two numbers entered by the user.
3. Write a program to convert temperature from Celsius to Fahrenheit and vice versa.
4. Write a program to calculate the factorial of a given number.
5. Write a program to check whether a number entered by the user is prime or not.
6. Write a program to print a pyramid pattern using asterisks.
7. Write a program to find the sum of elements in an array.
8. Write a program to calculate the length of a string and concatenate two strings.
9. Write a program to demonstrate pointer arithmetic operations.
10. Write a program to read data from a text file and display it on the screen.
11. Write a program to sort elements in an array using bubble sort.
12. Write a program to find the factorial of a number using recursion.

SEMESTER 3

CC-A3: COMPUTER NETWORKS

Course code	CC-A3			
Category	Core Course			
Course title	**Computer Network**			
Course ID	240/CS/CC301			
Scheme and Credits	L	T	P	Credits
	3	0	2	4
Theory Internal	25 marks			
Theory External	50 marks			
Practical Internal	5 marks			
Practical External	20 marks			
Total	100 marks			
Duration of Exam	3 hrs			

COURSE OBJECTIVES:

Aim of this course is to learn Computer Networks and its associated concepts and terminology along with the knowledge of Network architecture, design issues, and hardware components. Give exposure to the contemporary networking technologies and security issues for networks.

UNIT – I

Introduction to Computer Networks; Goals and applications; Types of Computer Networks; Network Design Issues and Protocols; Computer Communications and Networking Models; Communication Service methods and Data Transmission Modes; OSI Reference Model; OSI Service Types; Functions of layers of OSI Model; TCP/IP architecture; Purpose of major Protocols of TCP/IP;

UNIT – II

Physical layer: Analog and Digital Communication concepts; Copper Media; Fiber-Optic Media; Wireless Communications; Satellite Communication; Speed and Capacity of a communication channel; Network Hardware Components; Multiplexing; Switching; Dialup Networking; Analog Modem Concepts; DSL Service; Cable Modems; Leased lines; Home Networking Concepts;

UNIT – III

Data Link layer: Framing Techniques; Flow Control; Sliding Window Protocols; Error Control: Error Detection and Correction Methods; Medium Access Control: Random Access protocols; Token passing protocols; IEEE LAN Standards; Introduction to Wireless LANs;

UNIT – IV

Network layer: Routing Algorithms: Flooding; Shortest path Routing; Distance-Vector Routing; Link-State Routing; Multicast Routing; Techniques for Congestion Control; Network Security Issues: Security Goals; Threat Assessment; Network Attacks; Encryption Methods: Symmetric and Asymmetric-Key Ciphers; Firewalls, Digital Signatures, Authentication and Access Control Methods: Digital Certificates, Smart Cards, Kerberos;

Text Books:

[1] Michael A. Gallo, William M. Hancock, Computer Communications and Networking Technologies, CENGAGE learning.

[2] Behrouz A Forouzan, Data Communications and Networking, Mc-Graw Hill.

Reference Books:

[1] William Stallings, Data and Computer Communications, PHI.

[2] Andrew S. Tanenbaum, Computer Networks, PHI.

List of Experiments

1. Study of Parallel data Communication between two computers.
2. Study/Simulation of Network Topologies – Star, Bus & Ring
3. Implementation of stop and wait protocol using simulator.
4. Implementation of Sliding window protocol using simulator.
5. Implementation of Go-Back N protocol using simulator.
6. Implementation of Selective Repeat protocol using simulator.
7. Study the performance of the network with CSMA/ CD protocol.
8. Simulation/Implementation of 3 different routing algorithm

Semester 4

DATA STRUCTURE WITH C/C++

Course code	**CC-A**4			
Category	Core Subject			
Course title	Data Structure with C/C++			
Course ID	240/CS/CC401			
Scheme and Credits	L	T	P	Credits
	3	0	2	4
Theory Internal	25 marks			
Theory External	50 marks			
Practical Internal	5 marks			
Practical External	20 marks			
Total	100 marks			
Duration of Exam	3 hrs			

COURSE OBJECTIVES: Learning of data structure is like learning alphabets to learn any proper language. In this course student will be aware of memory management and use of data structure in computer programming.

UNIT – I

Data Structure Definition, Data Type vs. Data Structure, Categories of Data Structures, Data Structure Operations, Applications of Data Structures, Algorithms Complexity and Time-Space Trade-off, Big-O Notation. Strings: Introduction, Strings, String Operations, Pattern Matching Algorithms.

UNIT – II

Arrays: Introduction, Linear Arrays, Representation of Linear Array in Memory, Traversal, Insertions, Deletion in an Array, Multidimensional Arrays, Sparse Matrix. Algorithm for Insertion, Deletion Addition and Multiplication in 2- D Array. Searching and Sorting Techniques, Sorting Techniques: Bubble Sort, Merge Sort, Selection Sort, Heap Sort, Insertion Sort. Searching Techniques: Sequential Searching, Binary Searching, Search Trees.

UNIT – III

Stacks & Queues: Representation of Stacks, Stack Operations, Applications, Queues, Operations on Queues, Circular Queues, Dequeue, Priority Queues, Applications. Linked Lists: Introduction, Types, Operations (Insertion, Deletion, Traversal, Searching, Sorting), Applications, Dynamic Memory Management, Implementation of Linked Representations.

UNIT – IV

Trees: Basic Terminology, Representation, Binary Trees, Tree Representations using Linked List, Basic Operation on Binary tree, Traversal of Binary Trees: In order, Pre-order & Post-order, Applications of Binary tree. Algorithm of Tree Traversal with and without Recursion. Graphs: Definitions and Basic Terminologies, Representation of Graphs, Graph Traversals, Shortest Path Problem, Applications.

Text Books:

1. Seymour Lipschutz, Data Structures, Tata McGraw-Hill Publishing Company Limited, Schaum's Outlines.
2. Yedidyan Langsam, Moshe J. Augenstein, and Aaron M. Tenenbaum, Data Structures Using C, Pearson Education.

Reference Books:

1. Trembley, J.P. And Sorenson P.G., An Introduction to Data Structures With Applications, Mcgraw-Hill.
2. Mark Allen Weiss, Data Structures and Algorithm Analysis in C, Addison-Wesley.

List of Experiments

1. Write a program to implement Bubble Sort.
2. Write a program to implement Insertion Sort.
3. Write a program to implement Selection Sort.
4. Write a program to implement Heap Sort.
5. Write a program to implement Merge Sort.
6. Write a program to search an element using linear and binary search.
7. Write a program to implement stack operations.
8. Write a program to implement queue operations.
9. Write a program to implement linked list operations.
10. Write a program to perform various tree traversal algorithms for a given tree.

SEMESTER 5

CC-A5: OPERATING SYSTEM

Course code	CC-A5			
Category	Core Course			
Course title	**Operating System**			
Course ID	240/CS/CC501			
Scheme and Credits	L	T	P	Credits
	3	0	2	4
Theory Internal	25 marks			
Theory External	50 marks			
Practical Internal	5 marks			
Practical External	20 marks			
Total	100 marks			
Duration of Exam	3 hrs			

COURSE OBJECTIVES: The aim of the course is to provide knowledge of Operating System (OS) as a system program. Making student to learn about OS and linking OS as a powerful tool to make system work. Student will be able to learn types of OS and learn about system operations using OS.

UNIT – I

Introductory Concepts: Operating System Functions and Characteristics, Historical Evolution of Operating Systems, Operating System Structure and Operations; Types of Operating System: Real time, Multiprogramming, Multiprocessing, Batch processing; Operating System Services, Operating System Interface, Methodologies for Implementation of Operating System, Service System Calls, System Programs.

UNIT – II

Process Management: Process Concepts, Operations on Processes, Process States and Process Control Block. InterProcess Communication; Multithreaded Programming: Multithreading Models, Threading Issues; CPU Scheduling: Scheduling Criteria, Levels of Scheduling, Scheduling Algorithms, Multiple Processor Scheduling; Algorithm Evaluation.

UNIT – III

Synchronization: Critical Section Problem, Peterson's Solution, Synchronization Hardware, Semaphores, Classical Problem of Synchronization, Monitors, Atomic Transactions; Deadlocks: Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock 20 Avoidance, Deadlock Detection and Recovery.

UNIT – IV

Memory Management Strategies: Memory Management of Single-User and Multiuser Operating System, Partitioning, Swapping, Contiguous Memory Allocation, Paging and Segmentation; Virtual Memory Management: Demand Paging, Page Replacement Algorithms, Thrashing, Memory Mapped Files. Distributed Operating Systems: Types of Network based Operating Systems, Network Structure, Design Issues;

Text Books:

[1] Silberschatz A., Galvin P.B.,and Gagne G., Operating System Concepts, John Wiley & Sons. [2] Godbole, A.S., Operating Systems, Tata McGraw-Hill Publishing Company, New Delhi.

Reference Books:

[1] Deitel, H.M., Operating Systems, Addison-Wesley Publishing Company, New York.

[2] Tanenbaum, A.S., Operating System-Design and Implementation, Prentice Hall of India, New Delhi.

List of Practical:

1. Basics of UNIX
2. UNIX commands
3. Implementation of FCFS and SJF CPU scheduling algorithms
4. Implementation of Round Robin and Priority CPU Scheduling
5. Implementation of Producer-Consumer problem using semaphores
6. Implementation of FIFO Page Replacement Algorithms
7. Implementation of LRU Page Replacement Algorithms
8. Implementation of Sequential File Allocation Strategies
9. Implementation of Indexed File Allocation Strategies

Semester 6

SOFTWARE ENGINEERING

Course code	**CC-A6**			
Category	Core Subject			
Course title	**Software Engineering**			
Course ID	240/CS/CC601			
Scheme and Credits	L	T	P	Credits
	3	0	2	4
Theory Internal	25 marks			
Theory External	50 marks			
Practical Internal	5 marks			
Practical External	20 marks			
Total	100			
Duration of Exam	3Hrs			

COURSE OBJECTIVES: The objective of this course is to educate the students about the different models of software development and metrics used in software engineering.

UNIT – I

Introduction to Software Engineering, System Engineering Vs Software Engineering, Software Evolution, Software Characteristics, Components, Crisis–Problems and Causes, Software Feasibility, Software Process Models – VModel, Waterfall, Iterative Enhancement, Incremental, RAD, Prototyping, Spiral, Concurrent Development, Rational Unified Process & AGILE. Challenges in Software Engineering.

UNIT – II

Software Project Management – Planning, Scope Management, Cost Estimation – LOC, Function Point Analysis & COCOMO, Putnam Resource Allocation Model, Project Scheduling & Resource Management Gantt-Chart, PERT & CPM, Histogram, Team Building and Organization Charts, Project Monitoring & Risk Management, Software Configuration Management, Software Quality Assurance, Project Monitoring& Techniques. Software Requirement Analysis - Structured Analysis, Object

Oriented Analysis and Data Modelling, Software Requirement Specification, DFDs, Data Dictionaries, Decision Trees, Decision Tables & Structured English, ER Diagrams.

UNIT – III

Design and Implementation of Software – Basic Fundamentals, Design Methodology (Structured and Object Oriented), Design Approaches, User Interface Designing Tools & Techniques, Design Complexity, Monitoring and Control, Coding, Halstead's Software Science, McCabe's Cyclomatic Complexity Software Reliability: Metric and Specification, Fault Avoidance and Tolerance, Exception Handling, Defensive Programming, Component Based Development.

UNIT – IV

Software Testing – Fundamentals, Validation & Verification, White-Box and Black-Box Testing Techniques (Control Flow, Data Flow, Loop, Mutation, Load, Stress, Performance, Boundary Value, Equivalence Class, Decision Table, Cause Effect Graph Testing) Testing Strategies: Unit, Integration, Validation and System Testing, Alpha & Beta Testing, Debugging, Static Testing Strategies. Software & Maintenance: Maintenance Characteristics, Maintainability, Maintenance Tasks and Side Effects. Text

Text Books:

1. Gill, Nasib Singh. Software Engineering. Khanna Book Publishing Co. (P) Ltd., New Delhi.
2. Pressman S. Roger, Software Engineering, Tata McGraw Hill.
3. Jalote Pankaj, An Integrated Approach to Software Engineering, Narosa Publ. House.

Reference Books:

1. K. K. Aggarwal, Yogesh Singh, Software Engineering, New Age International.
2. Sommerville, Software Engineering, Pearson Education.
3. Fairley Richard, Software Engineering Concepts, Tata Mc-Graw Hill Ed. [4] Rajib Mall, Fundamentals of Software Engineering, PHI Learning.

List of Experiments:

1. Write down the problem statement for a suggested(any of your choice) system of relevance.
2. Do requirement analysis and develop Software Requirement Specification Sheet (SRS) for that suggested system.
3. To perform the function-oriented diagram: Data Flow Diagram (DFD) and Structured chart.
4. To perform the user's view analysis for the suggested system: Use case diagram.
5. To draw the structural view diagram for the system: Class diagram, object diagram.
6. To draw the behavioral view diagram: State-chart diagram, Activity diagram
7. To perform the behavioral view diagram for the suggested system: Sequence diagram, Collaboration diagram
8. To perform the implementation view diagram: Component diagram for the system.
9. To perform various testing using the testing tool unit testing, integration testing for a sample code of the suggested system.
10. Perform Estimation of effort using FP Estimation for chosen system.
11. To prepare time line chart/Gantt Chart/PERT Chart for any software project.