

B.Sc. Computer Science Model III

COURSE OUTCOMES

Semester 1

CS1CMT01 : Fundamentals of Digital Systems (Complementary)

CO1: Explain the different number systems , Understand the concepts of signed numbers, binary arithmetic, and digital codes such as BCD, gray code, and alphanumeric codes.

CO2: Design and implement logic gates using appropriate symbols and truth tables. Apply Boolean algebra laws and rules to simplify Boolean expressions and perform logic simplification using Karnaugh maps. Implement combinational logic circuits using basic gates and understand the functionality of various combinational logic components

CO3:Analyze logic waveforms and understand the behavior of logic levels in digital circuits. Analyze and evaluate different types of flip-flops, Analyze the operation of counters. Analyze the functionality and operation of shift registers

CO4: Evaluate the advantages and disadvantages of using NAND and NOR gates as universal gates in logic circuit design. Critically analyze the design and functionality of sequential circuits, including latches, flip-flops, and counters

EL1CMT05: Computer Fundamentals and Basics of PC Hardware (Complementary)

CO1: Understand the different generations and classifications of computers and their key characteristics.

CO2: Describe the organization and operation of various hardware components like SMPS,Inverter, UPS,DC regulated supply.

CO3: Assemble a basic PC by understanding the components of a CPU and motherboard and their functions.

CO4: Utilize various input and output devices for specific computing tasks.

CO5: Differentiate between primary memory and secondary memory and analyze the structure of a hard disk and its data storage mechanism.

CS1CRT02 : Methodology Of Programming And C Language

CO 1: Analyze the characteristics and factors influencing the selection of programming languages (Understand)

CO 2: Apply programming fundamentals including control structures, input/output functions, and decision statements in C (Apply)

CO 3: Demonstrate proficiency in using arrays, strings, pointers, and structures in C programming (Apply)

CO 4: Evaluate and differentiate between different types of functions, including recursion and storage classes in C (Evaluate)

CO 5: Design and implement programs using dynamic memory allocation and memory management techniques in C (Create)

CS1CRP01 : Software Lab - I

CO 1: Get comfortable with input and output functions using printf() and scanf().

CO 2: Write programs using decision-making and looping constructs.

CO 3: Perform tasks involving arrays and strings.

CO 4: Practice with pointers and dynamic memory allocation.

Semester 2

CS2CRT04 : Computer Organization and Architecture (Core)

CO 1: Understand the fundamental concepts of computer organization and design.. (Understanding)

CO 2: Apply the principles of central processing unit (CPU) organization and program control.

(Applying)

CO 3: Analyze the design and organization of memory hierarchies, including main memory, cache memory, and virtual memory. (Analysing)

CO 4: Evaluate the benefits and challenges of pipelining and vector processing in improving computer performance.(Evaluate)

CS2CRT05 - OBJECT ORIENTED PROGRAMMING USING C++-CORE

CO1: Apply the principles and concepts of object-oriented programming (OOP) in C++ to design and develop modular programs.(Applying)

CO2: Understand the concepts of constructors, destructors, and overloading techniques in C++ and their role in initializing objects and managing resources effectively. (Understanding)

CO3: Apply inheritance concepts in C++ to create class hierarchies and utilize different types of inheritance. (Applying)

CO4: Understand and recall the use of pointers, virtual functions, and polymorphism in C++ to achieve dynamic behavior and flexibility in object-oriented programming. (Remembering)

CS2CRT03 : Data Communication

CO 1: Analyze the differences between analog and digital data and signals, including their characteristics and transmission methods (Understand)

CO 2: Evaluate different types of transmission media, including guided and unguided media, and their suitability for different communication scenarios (Evaluate)

CO 3: Apply knowledge of digital transmission techniques such as analog-to-digital conversion, modulation, and multiplexing in communication systems (Apply)

CO 4: Assess the principles of analog transmission, including modulation techniques like ASK, FSK, PSK, and QAM, and their impact on bandwidth utilization (Evaluate)

CO 5: Analyze the concepts of switching in communication networks, including circuit-switched networks, packet switching, and virtual-circuit networks (Understand)

CS2CRP02 : Software Lab - II

CO 1: Write programs demonstrating default arguments and function overloading.

CO 2: Implement programs using array of objects, friend functions, and passing objects as arguments.

CO 3: Develop programs showcasing operator overloading (both binary and unary) using member and friend functions.

CO 4: Practice writing programs involving constructors, exploring various types of constructors.

CO 5: Create programs illustrating inheritance, including different types of inheritance.

Semester 3

CC3CRT01-DATABASE MANAGEMENT SYSTEMS

CO 1: Understand the fundamental concepts and advantages of the database

CO 2: Approach, including characteristics, users, and the three-schema architecture.

Apply entity-relationship modeling techniques to design and represent complex relationships in a database system.

CO 3: Implement SQL commands and queries to manipulate and retrieve data from a relational database, including data definition, basic queries, set operations, and complex queries.

CO 4: Analyze and apply normalization principles to ensure data integrity and optimize database schemas

CC3CRT02 : System Analysis And Design

CO 1: Describe information systems concepts and the role of business information systems in organizations. Explain the various levels of information systems and the tools used in system analysis and design.

CO 2 : Apply form design principles and tools like data flow diagrams to model information flow and processes. Evaluate the feasibility of proposed information systems projects based on cost and technology constraints.

CO 3 : Design comprehensive data dictionaries, system flow charts, and input/output components for efficient information systems. Propose innovative solutions to real-world system analysis and design challenges.

CO 4 : Identify and resolve issues related to system requirements and design constraints during the development life cycle.

CO 5 :Analyze the impact of changeover crises during the operation phase and recommend strategies for smooth system deployment.

EL3CMT08 : Networking Fundamentals

1. Describe various types of networks including LAN, WAN, and MAN, and analyze their characteristics and connections (Understand)

2. Evaluate different network topologies such as mesh, star, bus, and ring, and assess their suitability for various network architectures (Evaluate)

3. Apply knowledge of OSI and TCP/IP reference models to understand network protocols, addressing, and data link layer concepts (Apply)
4. Analyze error detection and correction techniques including block coding, cyclic redundancy check, and checksum in the data link layer (Analyze)
5. Evaluate routing and forwarding mechanisms, including distance vector routing and multicast routing protocols, and assess their impact on network performance (Evaluate)

CS3CRT08 Data Structures using C++ (Core)

- CO 1: Understand structured data fundamentals and its types, including arrays, in C++ (Understand).
- CO 2: Apply stack and queue implementations to solve problems like expression conversion in C++ (Apply).
- CO 3: Analyze linked list manipulation and memory management techniques in C++ (Analyze).
- CO 4: Synthesize advanced tree data structures and traversal algorithms in C++ (Create).
- CO 5: Evaluate file organization concepts and hashing techniques in C++ (Evaluate).

CC3CRP03 : Software Lab - III

Data Structures using C++.

- CO 1: Apply array operations, sorting, searching, and polynomial addition in C++.
- CO 2: Implement stack and queue data structures using arrays in C++.
- CO 3: Implement singly linked list data structure using dynamic memory allocation in C++.
- CO 4: Create and manipulate binary search trees including insertion and deletion of nodes in C++.

SQL

- CO 1: Create relational databases using DDL commands.
- CO 2: Apply DML commands to implement data manipulation such as insertion, updation, retrieval and deletion.
- CO 3: Construct complex SQL queries using joining tables.
- CO 4: Create views and simple stored procedures.

Semester 4

CS4CRT10 - LINUX ADMINISTRATION

CO1: Accumulate necessary knowledge and skills to navigate the Linux operating system, manage files and directories, and perform basic file operations efficiently.

CO2: Learn to perform system administration tasks efficiently in a Linux environment.

CO3: Understand the purpose, functionality, and use cases of servers.

CO4: Develop the skills to create shell programs for automating system tasks.

EL4CMT09: MICROPROCESSORS AND ASSEMBLY LANGUAGE PROGRAMMING (COMPLEMENTARY)

CO1: Understand the fundamental concepts and architecture of microprocessors. (Understanding)

CO2: Apply assembly language programming techniques using 8086 instructions. (Applying)

CO3: Implement advancement programming techniques in assembly language, including strings, procedures, and macros. (Evaluating)

CO4: Analyze the concepts of interrupts, timers, and direct memory access in microprocessors. (Analyzing)

CO5: Understand and recall the architecture and operation modes of advanced microprocessors, including the 80286, 80386, and 80486. (Remembering)

CC4CRP05: Assembly Language Programming Lab

CO 1: Write simple programs in assembly language.

CO 2: Write programs using conditional statements and looping constructs.

CO 3: Perform tasks involving arrays and strings.

CO 4: Write programs using macros, push and pop instructions.

CC4CRT03 : Computer Aided Optimization Techniques

1. Understand Operations Research (OR) principles and its application in decision making (Understand)
2. Formulate and solve Linear Programming Problems (LPP) using graphical and simplex methods (Apply)
3. Apply optimization techniques like transportation and assignment problems (Apply)
4. Evaluate network routing problems and scheduling techniques (Evaluate)
5. Integrate probability into scheduling using PERT method (Evaluate)

CS4CRT11: Web Programming Using PHP

CO 1: Understand the fundamental concepts and components of web development.

CO 2: Apply intermediate-level web development techniques

CO 3: Analyze and investigate web application functionalities

CO 4: Design and implement dynamic web applications with database integration

CC4CRP04 :SOFTWARE LAB IV

CO 1:Creating simple web pages using HTML tags and CSS.

CO 2: Apply client-side validations using javascript.

CO 3 : Create server-side scripts using PHP.

CO 4: Demonstrate database connectivity using PHP-MYSQL functions.

Semester 5

CS5CRT16 : Java Programming using Linux (Core)

- CO 1: Understanding Object-Oriented Programming in Java (Remember)
- CO 2: Mastering Java Language Constructs and Inheritance (Understand)
- CO 3: Exploring Java Packages, Exception Handling, and Multithreading (Apply)
- CO 4: Implementing Java Swing GUIs and Event Handling (Apply)
- CO 5: Building Java Applets and Database Connectivity with JDBC (Create)

CA5OPT02 : Computer Fundamentals, Internet and MS Office (Open Course)

- CO1: Develop an understanding about how a computer works, its components and its capabilities.
- CO2: Develop an understanding about the core component of any system: Operating system
- CO3: Gain an idea about internet technologies and e- world
- CO4: Develop an insight into the basics of computer networking concepts.
- CO5: Gain ability to create and edit documents, store and manage data in spreadsheets, and make and share presentations for various purposes.

CS5CRT15 : IT & Environment

1. Understand the multidisciplinary nature of environmental studies and the significance of public awareness. (Understand)
2. Analyze ecosystems, biodiversity, and conservation strategies, focusing on India's biodiversity richness. (Analyze)
3. Evaluate causes, effects, and mitigation measures of environmental pollution and disaster management. (Evaluate)

4. Assess social issues like urbanization, energy conservation, and environmental ethics, and legislative frameworks for protection. (Evaluate)
5. Explore IT's role in academia, including search techniques, educational software, and societal impacts. (Analyze)
6. Examine human rights principles and their intersection with environmental conservation. (Analyze)

CC5CRT04 - SYSTEM SOFTWARE AND OPERATING SYSTEM

CO 1: Understand the fundamental concepts of language processing, including lexical analysis, syntax analysis, and semantic analysis programming language grammar and classify different types of grammars. (Level: Understand)

CO 2: Understand the basics of operating systems, including process scheduling, inter-process communication, and CPU scheduling. (Level: Understand)

CO 3: Demonstrate the ability to analyze and select appropriate scheduling criteria and algorithms for various scenarios. (Level: Apply)

CS5CRP06 : Software Development Lab - I

CO 1: Develop Java programs utilizing classes and reading input from the keyboard.

CO 2 : Implement method overloading, method overriding, and inheritance in Java.

CO 3: Establish JDBC connections within Java programs.

CO 4: Incorporate exception handling mechanisms into Java programs.

Semester 6

CS6CRT19 - BIG DATA ANALYTICS

CO1: Possess the foundational knowledge required to navigate and analyze big data using appropriate tools and statistical methodologies.

CO2: Equipped to analyze and extract meaningful insights from data streams.

CO3: Generate the skills to work with Hadoop.

CO4: Gather knowledge and skills to apply Pig, Hive, HBase, and visualization techniques to effectively analyze and visualize big data for various applications.

CC6CBT01: Python and LaTeX

CO1: Understand the fundamentals of Python programming language, including variables, basic expressions, data types, and string operations.

CO2: Demonstrate proficiency in using control flow structures and manipulating basic data structures like lists, dictionaries, and tuples.

CO3: Develop the ability to create and utilize functions, both built-in and user-defined, with an understanding of function calls, parameters, returns, recursion, and scope concepts.

CO4: Acquire skills in handling files including reading and writing to files, interacting with directories, implementing interactive programming, and handling exceptions effectively.

CO5: Gain familiarity with basic LaTeX syntax and functionalities.

CS6CRT18 : Computer Graphics (Core)

CO 1: Understand the survey of Computer Graphics and the components of graphics systems.

CO 2: Apply line drawing algorithms such as DDA and Bresenham's line algorithm for graphics rendering.

CO 3: Apply 2D geometric transformations including translation, rotation, and scaling.

CO 4: Apply three-dimensional concepts and display methods, including polygon surfaces and constructive solid geometry methods.

CO 5: Implement computer animation techniques, including design principles, animation languages, and motion specifications.