UNIVERSITY OF NORTH BENGAL

Accredited by NAAC with grade "B++"

B.Sc. Computer Science 4-Year UNDERGRADUATE PROGRAM (FYUGP)
w.e.f. 2023-2024

Course Objectives & Outcomes for
B.Sc. Computer Science (Major & Minor) Under
THE NEW CURRICULUM AND CREDIT FRAMEWORK, 2022

B.Sc. Computer Science

UNIVERSITY OF NORTH BENGAL
RAJA RAMMOHANPUR, DARJEELING
WEST BENGAL
PIN-734013

COURSE OUTCOME – UNDERGRADUATE COMPUTER SCIENCE HONOURS (NEP SYSTEM) [SEM-1 to SEM-4]

NAME OF THE PROGRAMME	YEAR OF INTRODUCTION	COURSE OUTCOME		
BSC	2023-24	**COURSE NAME**	**TOPIC**	**COURSE OUTCOME**
		SEMESTER – 1 UCMSMAJ11001: (Programming in C)	Unit 1: Introduction to C	Students will learn basic concepts of programming language and the evolution of C
			Unit 2: Understanding Compilation and Execution in C	Students will get the idea of the compilation and execution process of the C language.
			Unit 3: Data Types, Variables, Constants, Operators, and Basic I/O	The students will learn the basic building blocks of the C language, such as data types, constants, various kinds of operators, and basic I/O operations.
			Unit 4: Expressions, Conditional Statements, and Iterative Statements	The students will learn about conditional and iterative statements, as well as how to write them in C.
			Unit 5: Understanding Functions	In this section, students will learn about the concepts of libraries and user-defined functions. They will understand the difference between these two types of functions and how to create and call them in their programs. Additionally, they will explore the advantages of using functions to modularize their code and make it more organized and reusable.
			Unit 6: Implementation of Arrays and Strings	Students will understand how to store and manipulate data efficiently using arrays and strings, which are fundamental data structures in computer programming.
			Unit 7: User-defined Data Types (Structures and Unions)	Structures and Unions are important concepts in programming that allow for the creation of custom data types. By understanding these concepts, students will be able to create more complex and efficient programmes.
			Unit 8: Pointers and References in C	Pointers and references are powerful tools in C programming that allow for efficient memory management and manipulation of data. By mastering these concepts, students will be able to write more complex and efficient programs.
			Unit 9: Memory Allocation in C	Students will learn how to use functions such as malloc() and free() to allocate and release memory, as well as the importance of managing memory efficiently to avoid memory leaks and other issues.

			Unit 10: File I/O	Students learn how to read and write data to files using various functions such as fopen(), fclose(), fprintf(), fscanf(), etc. They also learn about file modes, error handling, and binary file operations.
			Unit 11: Preprocessor Directives	Students will learn about preprocessor directives such as #define, #include, and #ifdef , and how to use them to enhance the functionality of their code and make it more efficient.
		SEMESTER – 1: **[UCMSMAJ11001L]** **(FM 20)**	Programming Fundamentals using C Lab	In Programming Fundamentals using C Lab, students will learn how to write and debug C programs, as well as how to use fundamental programming concepts such as variables, data types, control structures, and functions. They will also gain hands-on experience in writing and debugging C programs to solve various problems.
		SEMESTER – 1 **UCMSMAJ12002** **[Discrete Structures]** **(Theory)(FM 60)**	Unit 1: Introduction	To learn the concept of set, relation, and their types, Permutation-Combination, and Mathematical Induction.
			Unit 2: Growth of Functions	To get familiar with different mathematical asymptotic notations, summation formulas, and properties, etc
			Unit 3: Recurrences	To learn recurrence relations, linear recurrence relations, generating functions, etc.
			Unit 4: Graph Theory	To get familiar with basic terminology of graph theory, different models of graph and their representation, graph connectivity, tree, and spanning trees.
			Unit 5: Prepositional Logic	To understand logic connectives, WFF, Tautologies, and Inference theory.
		SEMESTER-1 **UCMSSEC11001** **(System Tools &** **Peripheral and Office** **Automation)**	Unit 1: Introduction to Computers	Upon completion, students will be able to explain the fundamental concepts of computing, including computer characteristics, generations, and architecture, and differentiate between various software types and programming language classifications (low-level vs. high-level, interpreters vs. compilers).
			Unit 2 : Peripheral devices	Upon successful completion of this course, students will be able to identify, describe, and differentiate between various input/output devices, primary and secondary memory technologies, and motherboard components used in computer systems.
			Unit 3 : Working with MS OFFICE	After completing this course, students will be able to create, format, and manage professional text documents using Microsoft Word. They will gain proficiency in text manipulation, integrating visual elements (tables, images, charts), and utilizing advanced features like Mail Merge and Spelling/Grammar Check

				Unit 4 : Working with MS Excel	Upon completion, students will be able to effectively format and manipulate Excel workbooks, perform calculations using functions, and create charts to analyze and visually present data.
				Unit 5 : Working with MS Power Point	Upon completion of this course, students will be able to create, format, and deliver professional presentations using Microsoft PowerPoint's core features, including slides, themes, animations, and the Slide Master.
			SEMESTER-1: UCMSSEC11001L (System Tools & Peripheral and Office Automation Lab)		After completing this course, students will be able to create, format, and manage documents in MS Word, utilizing features like mail merge and macros. They will also be capable of handling data in MS Excel, applying various formulas, number formats, and creating appropriate charts for presentation.
			SEMESTER-1: UCMSMDC11001 (Fundamentals of Data Science)	Unit 1: Data Governance	Upon completion of this course, students will be able to explain data governance and ethical data handling principles, and acquire, process, and analyze diverse datasets using foundational programming tools like Python and/or R.
				Unit 2: Exploratory data analysis	Upon completion, students will be able to perform fundamental data analysis, including univariate and bivariate analysis, utilize multivariate statistical techniques for high-dimensional data visualization, and apply data cleaning and tidying techniques.
				Unit 3: Classification Algorithms	By the end of this unit, students will be able to understand the fundamentals of supervised learning and decision trees. They will be able to generate, validate (using K-fold cross-validation), and implement decision tree models using Python or R.
				Unit 4: Regression	Upon successful completion of this course content, students will be able to apply the least squares method to fit linear regression models and implement the fitting algorithm using statistical programming languages like Python or R.
				Unit 5: Clustering	Upon completion, students will be able to explain the principles of unsupervised learning, describe the working of the K-Means clustering algorithm, and implement K-Means using Python or R.
			SEMESTER – 2 UCMSSEC12002: Digital Electronics (Theory)(FM 40)	Unit 1: Number system and codes	Learns different types of Number System, used in computer system and their representation types. Different basic arithmetic operations of number that are done within computer system are also learned.
				Unit 2: Basic logic circuits	To have concept of digital electronic logic gates and Boolean algebra and different representation of Boolean functions.

				Unit 3: Logic Families	Learns different logic families and their comparatives.
				Unit 4: Combinational Logic	Learn the structure and working process of different Combinational circuits like ADDER, DECODER, MUX etc.
				Unit 5: Sequential Circuit	Get familiar with the structure and working process of different sequential circuits like flip-flops, counters, registers etc.
		SEMESTER – 2 UCMSSEC12002L: Digital Electronics (Practical) (FM 20)			Introducing students' experimental learning, designing and implementation of different logic circuits are done using different ICs and trainer kit.
		SEMESTER-2: UCMSMDC12002 (Web Technology)		Unit 1: Introduction to WWW	By the end of this course, students will be able to explain the core architecture and fundamental components of the Internet, including protocols and addressing. They will also demonstrate the ability to apply web design principles for planning and developing functional websites and applications.
				Unit 2: Introduction to HTML	After completing this course, students will be able to describe the history and fundamental structure of HTML documents and utilize basic HTML tags for text formatting, linking, and list creation.
				Unit 3: Links, Images, and Tables	Upon completion of this unit, learners will be able to create fully linked and structured web pages using relative and absolute links, implement images (foreground and background), and effectively organize data using styled tables.
				Unit 4: HTML Frames and Forms	Upon completion of this course, students will be able to implement HTML frames and design and style complex HTML forms using various input types, attributes, and CSS, while understanding the differences between GET and POST methods for form data submission and redirection.
				Unit 5: Style Sheets	Upon completing this course, students will be able to describe the need for CSS and apply fundamental CSS syntax to style web page elements, including managing layouts, backgrounds, text, and borders.
				Unit 6: JavaScript	Upon completing this course, students will be able to design and implement interactive, dynamic client-side web pages using JavaScript fundamentals, including variables, functions, control structures, and object manipulation. They will also demonstrate proficiency in Document Object Model (DOM) manipulation and applying JavaScript, HTML, and CSS (DHTML) to

				handle web browser events, forms, and validation.
		SEMESTER – 2 UCMSMDC12003 :Software Development(Theory)(FM 60)	Unit 1: Introduction	To gain knowledge on the basics of software and its development process models.
			Unit 2: Software Requirements Analysis & Specifications	To learn the process of gathering and analysis of user requirements for a software to be developed.
			Unit 3: Software Project Management	To learn the management system required to deploy during the development of a software product.
			Unit 4: Software Design	To understand all the design issues of a software architecture.
			Unit 5: Testing strategies	To learn the process of testing a software product, after development and during maintenance, to find errors and weak points to achieve perfection.
			Unit 6: Quality Management	To learn how to develop and maintain reliable, efficient, and quality software.
		SEMESTER – 3 UCMSMAJ23003: Computer System Architecture (Theory) (FM 60)	Unit 1: Register transfer and micro operations	Acquires knowledge on different types of computer registers and different types of micro operations performed by a computer using registers.
			Unit 2: Basic Computer Organization and Design	Understands basic architecture of the computer system and some details on peripheral devices.
			Unit 3: Central Processing Unit	Learns computer instructions, assembly language, machine language, and different architectural designs of C.P.U.
			Unit 4: Computer Arithmetic	Learns how different arithmetic operations are done in a computer by implementing algorithms.
			Unit 5: Memory and Input-Output Organization	Gains knowledge on different types of memory, their hierarchy, and different I/O modules.
		SEMESTER – 3 UCMSMAJ23004: [Operating System] (theory) (FM 40)	Unit 1: Introduction	Learns basic OS functions, types of OS, and their working principles.
			Unit 2: Operating System Organization	Understands user mode, kernel mode of the OS, system call etc.
			Unit 3: Process Management	Learns the concept of process, threads, process scheduling, and related algorithms.
			Unit 4: Concurrency and Synchronization	Acquires knowledge on Inter-process communication, semaphores, and Deadlock.
			Unit 5: Memory Management	Understands different memory types, memory allocation strategies, partitioning, paging, and segmentation, virtual memory concept.

		SEMESTER – 3 CC 6 PRACTICAL UCMSMAJ23004 (FM 20)		Learns to implement different OS algorithms in C programming language to solve example programs.
		SEMESTER – 3 **UCMSMAJ23005: Data** **communication and** **networking (theory)** **(FM 60)**	Unit 1: Data Communication Fundamentals and Techniques	The students in this section will learn about some mechanisms that are required for data transmission, like modulation, multiplexing, and encoding techniques.
			Unit 2: Networks, Switching Techniques, and Access Mechanisms	Students will learn about the different types of switching techniques used in computer networks, such as circuit switching, packet switching, and message switching.
			Unit 3: Data Link Layer Functions and Protocol	Students will learn about how data is transmitted and received between network devices. Students will also gain knowledge of the error detection and correction techniques used in this layer.
			Unit 4: Multiple Access Protocol and Networks	Students will explore different types of Multiple Access Protocols that allow multiple devices to communicate over a shared communication channel with each other.
			Unit 5: Networks Layer Functions and Protocols	In this module, students will learn about the different functions of the network layers, including routing and addressing. They will gain an understanding of how data is transmitted through the network layer and the role it plays in routing packets to their intended destinations.
			Unit 6: Transport Layer Functions and Protocols	In this section, students will learn about the various transport layer protocols, such as TCP and UDP, their differences, and how they are used to ensure reliable data transmission over the network. They will also explore the functions of the transport layer, including flow control, congestion control, and error detection and correction.
			Unit 7: Overview of Application Layer Protocols	This segment will cover the various types of application layer protocols used in computer networks, including HTTP, FTP, SMTP, and DNS. Students will also gain an understanding of how these protocols enable communication between different devices and applications on a network.
		SEMESTER-3 **UCMSSEC23003** **(Data structure through C** **or C++)**	Unit 1: Arrays	Upon completing this content, students will be able to implement and manipulate single and multi-dimensional arrays, and represent sparse matrices efficiently using both array and linked list data structures.
			Unit 2: Stacks	Upon completion, students will be able to implement and manipulate stacks using both array and linked list representations. They will demonstrate proficiency in performing Push/Pop operations and applying stacks to

				solve problems involving prefix, infix, and postfix expressions.
			Unit 3: Linked Lists	Students will be able to implement, utilize, and perform fundamental operations (insertion and deletion) on various linked list structures, including singly, doubly, and circular lists. This includes understanding their representations and how they relate to data structures like stacks.
			Unit 4: Queues	The student will be able to analyze and implement Queue, De-queue, and Priority Queue data structures using both array and linked list representations, and perform their fundamental operations.
			Unit 5: Recursion	The course outcome is: Learners will be able to define and implement recursive solutions for simple problems, and evaluate the benefits and drawbacks of using recursion in programming.
			Unit 6: Trees	Students will be able to analyze, implement, and perform various operations (insertion, deletion, traversal) on fundamental tree data structures, including Binary Search Trees (BSTs) and Height-Balanced Trees (AVL Trees), to solve algorithmic problems efficiently.
			Unit 7: Searching and Sorting	The course outcome is: Students will be able to distinguish between linear and binary search algorithms, apply fundamental sorting techniques like Selection, Bubble, and Insertion Sort, and analyze their relative efficiencies.
		SEMESTER-3: **UCMSSEC23003** **(Data structure through C or C++ Lab)**		Students will be able to implement and compare fundamental data structures like arrays, linked lists (singly, doubly, circular), Stacks, and Queues. They will gain proficiency in programming search and sort algorithms (Linear, Binary, Insertion, Bubble, Selection) and understand the application of recursion and iteration for problem-solving.
		SEMESTER - 4 **UCMSMAJ24006** **[Database Management System]** **(Theory-FM 40)**	Unit-1 :Introduction	To gain basic knowledge of database systems and database management systems software.
			Unit 2: Database models	Introduces three basic database models and their comparison.
			Unit 3: Database Design, ER-Diagram	Learns designing database schemas based on conceptual modelling tools such as ER Diagrams. Understands database normalization and decomposition as a normalization tool.
			Unit-4: Transaction processing	To apply and relate the concept of transaction, concurrency control, and recovery in a database.

			Unit 5: SQL and Oracle	Acquire knowledge of SQL and ORACLE as a database query language to perform operations on the database.
		SEMESTER - 4 UCMSMAJ24006L [Database Management System] PRACTICAL (FM-20)		To use a desktop database package and use SQL to create, populate, maintain, and query a database.
		SEMESTER - 4 UCMSMAJ24007 [SOFTWARE ENGINEERING] (Theory-FM 60)	Unit 1: Introduction	To gain knowledge on the basics of software, software quality, and characteristics.
			Unit 2: Software development models	Understands different software development life cycle models.
			Unit 3: Software Requirements Analysis & Planning	Learns the process of gathering and analysis of user requirements for a software to be developed, then estimation for development, and basic knowledge on ERD, DFD.
			Unit 4: Software Project Management	To learn the management system required to deploy during the development of a software product.
			Unit 5: Coding	Learns different coding-related issues to develop the software.
			Unit 5: Unit 6: Testing Strategies & Tactics	To learn the process of testing a software product, after development and during maintenance, to find errors and weak points to achieve perfection.
		SEMESTER-4 UCMSMAJ24005 (Programming in Java)	Unit 1: Introduction to Java	Upon completion of this unit, students will be able to explain Java's architecture and core features, write, compile, and execute basic Java programs utilizing fundamental concepts like data types, operators, and control flow, and define and invoke Java methods.
			Unit 2: Arrays, Strings, and I/O	After completing this course content, the student will be able to design, implement, and manipulate both single and multi-dimensional arrays and String objects in Java, as well as manage basic input/output operations using streams for both console and file handling.
			Unit 3: Object-Oriented Programming Overview	By the end of this unit, students will be able to apply the principles of Object-Oriented Programming (OOP), effectively define and use classes, objects, and their members, and implement key OOP concepts like method overloading and class construction to solve programming problems.
			Unit 4: Inheritance, Interfaces, Packages, Enumerations, Autoboxing & Metadata	Upon completion of this unit, students will be able to implement core object-oriented concepts like inheritance, method overriding, and polymorphism, and effectively

				utilize interfaces, packages, and standard Java libraries while working with wrapper classes and enumerations.
			Unit 5: Exception Handling and Threading	By the end of this unit, students will be able to manage run-time errors using exception handling mechanisms (built-in and custom) and implement concurrent programming using multithreading techniques including synchronization and communication.
			Unit 6: Applets and Event Handling	Understand the design and implementation of GUI applications using Java Applets and the Java Foundation Classes (JFC), specifically leveraging AWT and Swing components for graphics, user interaction, and multimedia integration.
		SEMESTER-4 UCMSMAJ24005 (Programming in Java)		Upon completing this unit, students will be able to design and implement fundamental Java programs incorporating basic control structures, array and string manipulation, object-oriented programming (OOP) principles, and simple matrix operations.
		SEMESTER-1 and SEMESTER-2 UCMSMIN10001 (Fundamentals of Computers & Problem Solving)	Unit 1: Knowing Computer Basics	Students will be able to explain the evolution, generations, and classification of computers, and describe the fundamental hardware concepts, including I/O devices, processing components, memory, storage, and connectivity architecture.
			Unit 2: Software and Data Fundamentals	Upon completion, students will be able to classify various types of software and explain the function of system tools (OS, loaders, linkers, debuggers, translators), and differentiate between data, information, and database concepts, including units of measurement and language generations.
			Unit 3: DOS (Disk Operating System)	Understand the fundamentals of the Disk Operating System (DOS), including its file and directory structure. Master basic DOS commands for file management, system control, and directory navigation. Explain the booting process (warm/cold boot) and the role of BIOS.
			Unit 4: Applications of computers	Students will be able to identify and describe the diverse applications of computer technology across major professional domains and real-world scenarios, including business, industry, healthcare, education, and finance. This outcome focuses on recognizing the computer's role in data management, process control, communication, and decision support.

			Unit 5: Problem-solving and programming	Students will be able to analyze and represent problem-solving steps using algorithms and flowcharts. They will also distinguish between different program structures and programming language types.
		SEMESTER-3 and SEMESTER-4: UCMSMIN20002 (Programming in C)	Unit 1: Introduction to C	Upon completion of this unit, students will be able to describe the history of the C programming language, and develop, compile, and execute basic C programs using the main() function.
			Unit 2: Data Types, Variables, Constants, Operators and Basic	Upon completion of this unit, students will be able to apply fundamental C programming constructs including variables, data types, operators, and I/O functions to write, compile, and execute basic programs. This includes understanding variable scope and the use of named constants and comments.
			Unit 3: Expressions, Conditional Statements and Iterative Statements	Students will be able to construct and evaluate C expressions using various operators and implement control flow in programs using conditional and iterative statements, including nesting and loop control keywords.
			Unit 4: Functions and Arrays	Upon completing this unit, students will be able to design and implement programs using various types of functions (including call by value/reference and returning values) and effectively utilize both one-dimensional and two-dimensional arrays for data storage and manipulation.
			Unit 5: Derived Data Types	The student will be able to declare, initialize, and effectively use C structures and unions, including arrays of structures and nested structures, to organize and manipulate complex data.
		SEMESTER-3 and SEMESTER-4 : UCMSMIN20002 (Programming in C Tutorial)		Upon successful completion of this unit, students will be able to design and implement basic programs using foundational data types and control structures (like loops and conditionals) to solve simple computational problems such as number manipulation, series generation, and fundamental arithmetic calculations.