NCCF-2023-24

National Curriculum and Credit Framework (NCCF)

Syllabus

for

Major, Minor, Multidisciplinary and Skill Enhancement Courses in

Computer Science

w.e.f. Academic Session 2023-24

Kazi Nazrul University
Asansol, Paschim Bardhaman
West Bengal 713340

Semester- I

MAJOR COURSE

Course Name: Introduction to Programming using C

Course Code: BSCCOSMJ101

Course Type: **Major (Theoretical & Practical)**	Course Details: **MJC-1**			L-T-P: **3 – 0 – 4**	
Credit: **5**	Full Marks: **100**	CA Marks		ESE Marks	
		Practical	Theoretical	Practical	Theoretical
		30	**15**	**20**	**35**

Course Content:

Theory

UNIT I. Introduction to computers, Evolution, Generation of Computers, Computers Hierarchy, Different components of computer (CPU, ALU, different types of memory etc.), Number System – Binary, Hexa, Octal, BCD System, Introduction to operating environment.

UNIT II. Introduction to Programming, Program Concept, Characteristics of Programming, Stages in Program Development, Algorithms, Notations, Flowcharts, Types of Programming Methodologies, Introduction to C Programming - Basic Program Structure in C, Variables and Assignments, Input and Output, Selection and Repetition Statements.

UNIT III. Top-Down Design, Predefined Functions, Programmer-defined Function, Local Variable, Recursion - Developing Recursive Definition of Simple Problems and their implementation.

UNIT IV. Introduction to Arrays, Declaration and Referring Arrays, Arrays in Memory, Initializing Arrays.Arrays in Functions, Multi-Dimensional Arrays, Searching in Array.

UNIT V. Pointers - Simple use of Pointers (Declaring and Dereferencing Pointers to simple variables), Pointers to Pointers, Call-By-Value and Call-By-Reference Parameters.

UNIT VI. Structures - Member Accessing, Pointers to Structures, Structures and Functions, Arrays of Structures, Unions.

UNIT VII. Strings - Declaration and Initialization, Reading and Writing Strings, Arrays of Strings, String and Function, Strings and Structure, Standard String Library Functions.

UNIT VIII. File Handling – File opening modes, use of files for data input and output. merging and copy files.

Practical

UNIT I. Given the problem statement, students are required to formulate problem, develop flowchart/algorithm, write code, execute and test it. Students should be given assignments on following:

a) To learn elementary techniques involving arithmetic operators and mathematical expressions, appropriate use of selection (if, switch, conditional operators) and control structures.

b) Learn how to use functions and parameter passing in functions, writing recursive programs.

UNIT II. Students should be given assignments on following:

a) Write Programs to learn the use of strings and string handling operations.

b) Problems which can effectively demonstrate use of Arrays. Structures and Union.

c) Write programs using pointers and functions.

d) Write programs to use files for data input and output.

Internal (CA) Evaluation: Practical Note Book (15 marks), Two experiments (10 marks) – one from each unit, Viva-voce (5 marks)

ESE Evaluation: Two experiments (10 marks) – one from each unit, Viva-voce (10 marks)

References/ Suggested Readings:

1. Problem Solving and Program Design in C, J. R. Hanly and E. B. Koffman, Pearson.
2. C Programming, Karnighan & Ritchie, PHI
3. Programming through C, Richard Johnsonbaugh and Martin Kalin, Pearson Education
4. Programming in C, B.S. Gottfried, Sahaum Series.
5. Programming in ANSI C, E. Balaguruswami, TMH

MINOR COURSE

Course Name: Introduction to Programming using C

Course Code: BSCCOSMN101

Course Type: **Minor (Theoretical & Practical)**	Course Details: **MNC-1**			L-T-P: **3 – 0 – 4**	
Credit: **5**	Full Marks: **100**	CA Marks		ESE Marks	
		Practical	Theoretical	Practical	Theoretical
		30	**15**	**20**	**35**

Course Content:

Theory

UNIT I. Introduction to computers, Evolution, Generation of Computers, Computers Hierarchy, Different components of computer (CPU, ALU, different types of memory etc.), Number System – Binary, Hexa, Octal, BCD System, Introduction to operating environment.

UNIT II. Introduction to Programming, Program Concept, Characteristics of Programming, Stages in Program Development, Algorithms, Notations, Flowcharts, Types of Programming Methodologies, Introduction to C Programming - Basic Program Structure in C, Variables and Assignments, Input and Output, Selection and Repetition Statements.

UNIT III. Top-Down Design, Predefined Functions, Programmer-defined Function, Local Variable, Recursion - Developing Recursive Definition of Simple Problems and their implementation.

UNIT IV. Introduction to Arrays, Declaration and Referring Arrays, Arrays in Memory, Initializing Arrays.Arrays in Functions, Multi-Dimensional Arrays, Searching in Array.

UNIT V. Pointers - Simple use of Pointers (Declaring and Dereferencing Pointers to simple variables), Pointers to Pointers, Call-By-Value and Call-By-Reference Parameters.

UNIT VI. Structures - Member Accessing, Pointers to Structures, Structures and Functions, Arrays of Structures, Unions.

UNIT VII. Strings - Declaration and Initialization, Reading and Writing Strings, Arrays of Strings, String and Function, Strings and Structure, Standard String Library Functions.

UNIT VIII. File Handling – File opening modes, use of files for data input and output. merging and copy files.

Practical

UNIT I. Given the problem statement, students are required to formulate problem, develop flowchart/algorithm, write code, execute and test it. Students should be given assignments on following:

a) To learn elementary techniques involving arithmetic operators and mathematical expressions, appropriate use of selection (if, switch, conditional operators) and control structures.

b) Learn how to use functions and parameter passing in functions, writing recursive programs.

UNIT II. Students should be given assignments on following:

a) Write Programs to learn the use of strings and string handling operations.

b) Problems which can effectively demonstrate use of Arrays. Structures and Union.

c) Write programs using pointers and functions.

d) Write programs to use files for data input and output.

Internal (CA) Evaluation: Practical Note Book (15 marks), Two experiments (10 marks) – one from each unit, Viva-voce (5 marks)

ESE Evaluation: Two experiments (10 marks) – one from each unit, Viva-voce (10 marks)

References/ Suggested Readings:

1. Problem Solving and Program Design in C, J. R. Hanly and E. B. Koffman, Pearson.
2. C Programming, Karnighan & Ritchie, PHI
3. Programming through C, Richard Johnsonbaugh and Martin Kalin, Pearson Education
4. Programming in C, B.S. Gottfried, Sahaum Series.
5. Programming in ANSI C, E. Balaguruswami, TMH

MULTIDISCIPLINARY COURSE

Course Name: Information and Communication Technology

Course Code: MDC117

Course Type: **MD (Theoretical)**	Course Details:**MD-1**			L-T-P: **3–0–0**	
Credit: **3**	Full Marks: **50**	CA Marks		ESE Marks	
		Practical	Theoretical	Practical	Theoretical
			15		**35**

Course Content:

Theory

UNIT I. Introduction to ICT: Definition of ICT and its significance in today's world, Historical evolution of ICT and major milestones,ICT's role in Education, ICT's role in various industries and societal impacts.

UNIT II. Computer Systems and Architecture: Overview of computer hardware components, CPU, memory, storage devices, and peripherals, Basics of computer architecture and operating systems.

UNIT III. Software and Programming Fundamentals: Introduction to software types: system software, application software, and programming languages, Basics of algorithm design and problem-solving techniques.

UNIT IV. Data Communication and Networks: Understanding data transmission and communication protocols, Introduction to network topologies and architectures, Local Area Networks (LANs), Wide Area Networks (WANs), and the Internet.

UNIT V. Basics of Email: Traditional mail vs Email, Understanding of email addresses, setting up your own email account, Email providers,E-mail protocols, Format of an E- mail message, Description of E-mail Headers, E- mail contents and encoding, E-mail clients.

UNIT VI. Emerging Technologies: The importance of information security in ICT, Cloud Computing and Virtualization: Understanding cloud computing models (SaaS, PaaS, IaaS), Virtualization technologies and their role in cloud infrastructure, Evolution of mobile technologies and their impact on ICT, Exploration of cutting-edge technologies (e.g., Internet of Things, Artificial Intelligence, Virtual Reality).

UNIT VII. Ethical and Social Implications of ICT: Ethical considerations in the use of ICT and data privacy, ICT's impact on society, economy, and the environment.

References/ Suggested Readings:

1. P S Kawatra, Fundamentals of Information and Communication Technology (ICT), B.R. Publishing Corporation.
2. C. Thatchinamoorthy, Fundamentals of Information Communication Technology, Notion Press.

SKILL ENHANCEMENT COURSE

Course Name: Office Automation Software Lab

Course Code: BSCCOSSE101

Course Type: SEC **(Practical)**	Course Details: **SEC-1**			L-T-P: **0 – 0 – 6**	
Credit: 3	Full Marks: **50**	CA Marks		ESE Marks	
		Practical	Theoretical	Practical	Theoretical
		30		**20**	

Course Content:

Practical

UNIT I. Windows Basics: Introduction of windows OS, navigating the Windows 10 user interface, Creating accounts in Windows, Opening apps and programs, working with files, using the Start button and Start menu, Accessing and using the Action Center, Working with apps and programs on the taskbar, Customizing settings in Windows 10, including backgrounds, screensavers, and more, Using the Settings app and the Control Panel.

UNIT II. MS Word and Google Docs: Overview, creating, saving, opening, importing, exporting, and inserting files, formatting pages, paragraphs and sections, indents and outdents, creating lists and numbering. Headings, styles, fonts and font size, editing, positioning, viewing texts, searching and replacing text, inserting page breaks, page numbers, bookmarks, symbols, and dates. Using tabs and tables, header, footer, and printing,

UNIT III. MS Excel and Google Sheets: Worksheet overview, entering information, worksheet creation, opening and saving workbook, formatting numbers and texts, protecting cells, producing charts, and printing operations. Application of Excel for obtaining statistical parameters, Mean, Median, Mode, average, co-relation, Regression, Data capturing using Google Forms.

UNIT IV. MS PowerPoint or Google Slides: Slide creation with PowerPoint, Presenting shows for corporate and commercial using PowerPoint.

UNIT V. Graphics and Image Editing Software: Overview of graphic design and image editing applications (e.g., Adobe Photoshop, GIMP), Understanding basic image editing techniques (e.g., cropping, resizing, retouching), Creating and manipulating graphics for various purposes.

UNIT VI. Web Browsing and Internet Applications: Navigating web browsers and utilizing essential features, Understanding internet protocols and security considerations, Exploring common internet applications (e.g., email clients, cloud storage, online collaboration tools).

UNIT VII. File Compression and Archiving Software: Introduction to file compression formats (e.g., ZIP, RAR), Compressing and decompressing files and folders, Managing archived files and folders.

Internal (CA) Evaluation: Practical Note Book (15 marks), One experiment (10 marks), Viva-voce (5 marks).

ESE Evaluation: One experiment (10 marks), Viva-voce (10 marks).

References/ Suggested Readings:

1. Introduction to Computers with MS-Office, Leon, TMH.
2. Learn Microsoft Office 2019, Linda Foulkes, HP.

Semester- II

MAJOR COURSE

Course Name: Data Structures and Algorithms

Course Code: BSCCOSMJ201

Course Type: **Major (Theoretical & Practical)**	Course Details: **MJC-2**			L-T-P: **3 – 0 – 4**	
Credit: **5**	Full Marks: **100**	CA Marks		ESE Marks	
		Practical	Theoretical	Practical	Theoretical
		30	**15**	**20**	**35**

Course Content:

Theory

UNIT I. Basic concepts- Data, Data Structures, ADT, Algorithm Specification-Introduction, Recursive algorithms, Data Abstraction, Performance analysis, Linear and Non Linear data structures.

UNIT II. Singly Linked Lists - Operations, Concatenating, Circularly linked lists - Operations for Circularly linked lists, Doubly Linked Lists - Operations. Polynomial and sparse matrix representation using linked list.

UNIT III. Stack- Definition and Operations, Array and Linked Implementations, Applications - Valid Expression Checking (Parenthesis matching), Reversal of string, Infix to Postfix Conversion, Postfix Expression Evaluation, Recursion Implementation.

UNIT IV. Queue - Definition and Operations, Array and Linked Implementations, Applications, Circular Queues - Insertion and Deletion Operations, Priority Queue-Definition and Implementation, Dequeue (Double Ended Queue) - Introduction.

UNIT V. Searching Methods – Linear and Binary.

UNIT VI. Sorting Methods – Bubble, Insertion, Selection, Shell, Using Divide-Conquer Approach (Quick and Merge sort), Comparison of Sorting Methods.

UNIT VII. Trees, Representation of Trees, Binary tree, Properties of Binary Trees, Binary Tree Representations- Array and Linked Representations, Binary Tree Traversals, Threaded Binary Trees, Binary Search tree - Creation, Insertion, Deletion and Search, AVL tree-Definition, Examples, Insertion and Rotations, B tree, B+ tree, Heap- Definition, Min heap, Max heap, Insertion and Deletion. Priority Queue using Heap.

UNIT VIII. Graphs, Graph ADT, Graph Representations, Graph Traversals and Searching,

Practical

Students are required to write and practically execute programs to solve problem using various data structures. The teacher can suitably device problems which help students experiment using the suitable data structures and operations. Some of the problems are indicated below.

1. Write program that uses functions to perform the following:
   a) Creation of list of elements where the size of the list, elements to be inserted and deleted are dynamically given as input.
   b) Implement the operations, insertion, deletion at a given position in the list and search for an element in the list
   c) To display the elements in forward / reverse order
2. Write recursive programs for Factorial, Fibonacci numbers, Towers of Hanoi etc.
3. Write a program to implement stack (using array and linked list). Write a program that demonstrates the application of stack operations (Eg: infix expression to postfix conversion, postfix evaluation).
4. Write programs to implement queue using array and linked list.
5. Write program that implements linear (using array and linked list) and binary search.
6. Write programs of a) Bubble sort b) Insertion Sort c) Selection Sort d) Quicksort etc.
7. Write a program to create a Binary Search Tree and insertion and deletion of node from the tree. Write recursive and non-recursive routines to traverse a binary tree in preorder, inorder and postorder.

Internal (CA) Evaluation: Practical Note Book (15 marks), Two experiments (10 marks), Viva-voce (5 marks)

ESE Evaluation: Two experiments (10 marks), Viva-voce (10 marks)

References/ Suggested Readings:

1. Fundamentals of Data structures in C, 2nd Edition, E. Horowitz, S. Sahni and Susan Anderson-Freed, Universities Press.
2. Data structures and Algorithm Analysis in C, 2nd edition, M. A. Weiss, Pearson.
3. Data structures, Lipschutz: Schaum's outline series, Tata McGraw-Hill
4. Data Structure through C in Depth, S.K. Srivastava and Deepali Srivastava, B.P.B Publication.

MINOR COURSE

Course Name: Data Structures and Algorithms

Course Code: BSCCOSMN201

<table>
<tr><td>Course Type: <b>Minor (Theoretical & Practical)</b></td><td colspan="3">Course Details: <b>MNC-2</b></td><td colspan="2">L-T-P: <b>3 – 0 – 4</b></td></tr>
<tr><td rowspan="3">Credit: <b>5</b></td><td rowspan="3">Full Marks: <b>100</b></td><td colspan="2">CA Marks</td><td colspan="2">ESE Marks</td></tr>
<tr><td>Practical</td><td>Theoretical</td><td>Practical</td><td>Theoretical</td></tr>
<tr><td><b>30</b></td><td><b>15</b></td><td><b>20</b></td><td><b>35</b></td></tr>
</table>

Course Content:

Theory

UNIT I. Basic concepts- Data, Data Structures, ADT, Algorithm Specification-Introduction, Recursive algorithms, Data Abstraction, Performance analysis, Linear and Non Linear data structures.

UNIT II. Singly Linked Lists - Operations, Concatenating, Circularly linked lists - Operations for Circularly linked lists, Doubly Linked Lists - Operations. Polynomial and sparse matrix representation using linked list.

UNIT III. Stack- Definition and Operations, Array and Linked Implementations, Applications - Valid Expression Checking (Parenthesis matching), Reversal of string, Infix to Postfix Conversion, Postfix Expression Evaluation, Recursion Implementation.

UNIT IV. Queue - Definition and Operations, Array and Linked Implementations, Applications, Circular Queues - Insertion and Deletion Operations, Priority Queue-Definition and Implementation, Dequeue (Double Ended Queue) - Introduction.

UNIT V. Searching Methods – Linear and Binary.

UNIT VI. Sorting Methods – Bubble, Insertion, Selection, Shell, Using Divide-Conquer Approach (Quick and Merge sort), Comparison of Sorting Methods.

UNIT VII. Trees, Representation of Trees, Binary tree, Properties of Binary Trees, Binary Tree Representations- Array and Linked Representations, Binary Tree Traversals, Threaded Binary Trees, Binary Search tree - Creation, Insertion, Deletion and Search, AVL tree-Definition, Examples, Insertion and Rotations, B tree, B+ tree, Heap- Definition, Min heap, Max heap, Insertion and Deletion. Priority Queue using Heap.

UNIT VIII. Graphs, Graph ADT, Graph Representations, Graph Traversals and Searching,

Practical

Students are required to write and practically execute programs to solve problem using various data structures. The teacher can suitably device problems which help students experiment using the suitable data structures and operations. Some of the problems are indicated below.

8. Write program that uses functions to perform the following:
   a) Creation of list of elements where the size of the list, elements to be inserted and deleted are dynamically given as input.
   b) Implement the operations, insertion, deletion at a given position in the list and search for an element in the list
   c) To display the elements in forward / reverse order
9. Write recursive programs for Factorial, Fibonacci numbers, Towers of Hanoi etc.
10. Write a program to implement stack (using array and linked list). Write a program that demonstrates the application of stack operations (Eg: infix expression to postfix conversion, postfix evaluation).
11. Write programs to implement queue using array and linked list.
12. Write program that implements linear (using array and linked list) and binary search.
13. Write programs of a) Bubble sort b) Insertion Sort c) Selection Sort d) Quicksort etc.
14. Write a program to create a Binary Search Tree and insertion and deletion of node from the tree. Write recursive and non-recursive routines to traverse a binary tree in preorder, inorder and postorder.

Internal (CA) Evaluation: Practical Note Book (15 marks), Two experiments (10 marks), Viva-voce (5 marks)

ESE Evaluation: Two experiments (10 marks), Viva-voce (10 marks)

References/ Suggested Readings:

1. Fundamentals of Data structures in C, 2nd Edition, E. Horowitz, S. Sahni and Susan Anderson-Freed, Universities Press.
2. Data structures and Algorithm Analysis in C, 2nd edition, M. A. Weiss, Pearson.
3. Data structures, Lipschutz: Schaum’s outline series, Tata McGraw-Hill
4. Data Structure through C in Depth, S.K. Srivastava and Deepali Srivastava, B.P.B Publication.

MULTIDISCIPLINARY COURSE

Course Name: Social Media and Cyber Awareness

Course Code: MDC209

Course Type: **MD (Theoretical)**	Course Details:**MD-2**			L-T-P: **3–0–0**	
Credit: **3**	Full Marks: **50**	CA Marks		ESE Marks	
		Practical	Theoretical	Practical	Theoretical
			15		**35**

Course Content:

Theory

UNIT I. Introduction to Social Media: Overview of major social media platforms and their features, History and evolution of social media, Pros and cons of social media usage.

UNIT II. Understanding Digital Footprints and Privacy: What is a digital footprint?, The importance of protecting personal information online, Privacy settings on social media platforms.

UNIT III. Responsible Social Media Usage and Digital Citizenship: Building a positive online presence, Ethical considerations in social media communication, Understanding online reputation management.

UNIT IV. Social Media and Mental Health: The relationship between social media usage and mental well-being, Strategies for promoting a healthy balance between online and offline life, Dealing with social media addiction.

UNIT V. Cyber Security Fundamentals: Introduction to cyber security concepts, Common cyber threats and attacks, Password best practices and multi-factor authentication.

UNIT VI. Cyber bullying and Online Harassment: Defining cyber bullying and its impact on individuals, Strategies for preventing and dealing with cyber bullying, Supporting victims and fostering a respectful online environment.

UNIT VII. Identifying and Avoiding Online Scams: Recognizing common online scams and phishing attempts, Techniques used by scammers and how to stay vigilant, Reporting and dealing with online fraud.

UNIT VIII. Creating Your Online Safety Plan: Developing a personal cyber security and social media responsibility plan, Resources for staying informed about online safety.

References/ Suggested Readings:

1. P. W. Singer and Allan Friedman, Cyber Security and Cyber War, Oxford University Press.
2. Raef Meeuwisse, Cybersecurity For Beginners, Lulu Publishing Services.

SKILL ENHANCEMENT COURSE

Course Name: Basics of Python

Course Code: BSCCOSSE201

Course Type: SEC **(Practical)**	Course Details: **SEC-2**			L-T-P: **0 – 0 – 6**	
Credit: 3	Full Marks: **50**	CA Marks		ESE Marks	
		Practical	Theoretical	Practical	Theoretical
		30		**20**	

Course Content:

Practical

UNIT I. Introduction to Python, Python, Features of Python, Execution of a Python, Program, Writing Our First Python Program, Data types in Python. Python Interpreter and Interactive Mode; Values and Types: int, float, boolean, string, and list; Variables, Expressions, Statements, Tuple Assignment, Precedence of Operators, Comments; Modules and Functions, Function Definition and use, Flow of Execution, Parameters and Arguments

UNIT II. Operators in Python, Input and Output, Control Statements. Boolean Values and operators, Conditional (if), Alternative (if-else), Chained Conditional (if-elif-else); Iteration: state, while, for, break, continue, pass; Fruitful Functions: Return Values, Parameters, Local and Global Scope, Function Composition, Recursion

UNIT III. Arrays in Python, Strings and Characters. Strings: String Slices, Immutability, String Functions and Methods, String Module; Lists as Arrays. Illustrative Programs: Square Root, gcd, Exponentiation, Sum an Array of Numbers, Linear Search, Binary Search.

UNIT IV. Functions, Lists and Tuples. List Operations, List Slices, List Methods, List Loop, Mutability, Aliasing, Cloning Lists, List Parameters; Tuples: Tuple Assignment, Tuple as Return Value; Dictionaries: Operations and Methods; Advanced List Processing - List Comprehension; Illustrative Programs: Selection Sort, Insertion Sort, Merge sort, Histogram.

UNIT V. Files and Exception: Text Files, Reading and Writing Files, Format Operator; Command Line Arguments, Errors and Exceptions, Handling Exceptions, Modules, Packages; Illustrative Programs: Word Count, Copy File.

The students are required to verify their ability to use core programming basics and program design with functions using Python programming language. The teacher shall programs to strengthen the practical expertise of the students. The following is an indicative list of programs that can be practised.

1. Write a program to demonstrate different number data types in Python.
2. Write a program to perform different Arithmetic Operations on numbers in Python.
3. Write a program to create, concatenate and print a string and accessing sub-string from a given string.

4. Write a python script to print the current date in the following format “Sat Oct 11 02:26:23 IST 2020”
5. Write a program to create, append, and remove lists in python.
6. Write a program to demonstrate working with tuples in python.
7. Write a program to demonstrate working with dictionaries in python.
8. Write a python program to find largest of three numbers.
9. Write a Python program to construct the different pattern, using a nested for loop, Like

```
*
* *
* * *
* *
 *
```

10. Write a Python script that prints prime numbers less than 20.
11. Write a python program to define a module to find Fibonacci Numbers and import the module to another program.
12. Write a python program to define a module and import a specific function in that module to another program.
13. Write a program that inputs a text file. The program should print all of the unique words in the file in alphabetical order.
14. Write a Python class to convert an integer to a roman numeral.
15. Write a Python class to reverse a string word by word.

Internal (CA) Evaluation: Practical Note Book (15 marks), Two experiments (10 marks), Viva-voce (5 marks).

ESE Evaluation: Two experiments (10 marks), Viva-voce (10 marks).

References/ Suggested Readings:

1. Alex Martelli, Python in a Nutshell, Oreilly Publication.
2. Allen Downey, Think Python, Green Tea Press.
3. Wesley J. Chun, Core Python Programming, Pearson Education.
4. Mark Lutz, Learning Python, Oreilly Publication.
5. Kenneth A. Lambert, Fundamentals of Python: First Programs, Course Technology Inc.