National Curriculum and Credit Framework (NCCF)

Syllabus

for

Computer Science

w.e.f. Academic Session 2023-24

Kazi Nazrul University

Asansol, Paschim Bardhaman

West Bengal 713340

Semester- I

MAJOR COURSE - 1

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 - 1

Course Name: Introduction to Programming using C

Course Code: BSCCOSMN101

<table>
<tr><td rowspan="2">Course Type: Minor (Theoretical & Practical)</td><td colspan="3" rowspan="2">Course Details: MNC-1</td><td colspan="2">L-T-P: 3 – 0 – 4</td></tr>
<tr></tr>
<tr><td rowspan="3">Credit: 5</td><td rowspan="3">Full Marks: 100</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>30</td><td>15</td><td>20</td><td>35</td></tr>
</table>

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 - 1

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 - 1

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 - 2

Course Name: Data Structures and Algorithms

Course Code: BSCCOSMJ201

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

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 - 2

Course Name: Data Structures and Algorithms

Course Code: BSCCOSMN201

Course Type: **Minor (Theoretical & Practical)**	Course Details: **MNC-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.

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 - 2

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 - 2

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.

Semester- III

MAJOR COURSE - 3

Course Name: Discrete Mathematics

Course Code: BSCCOSMJ301

Course Type: **Major (Theoretical)**	Course Details: **MJC-3**			L-T-P: 4 – 1 – 0	
Credit: **5**	Full Marks: **100**	CA Marks		ESE Marks	
		Practical	Theoretical	Practical	Theoretical
		---	**30**	**---**	**70**

Course Content:

Theory

UNIT I. Sets: Finite and Infinite Sets, Uncountable Infinite Sets, problems based on set theory.

Functions: Domain, Co-domain, Range, Injective, surjective and bijective functions Equal function, Exponential function, Logarithmic function, Square function, Cube function, Relations: Reflexive, Symmetric, Anti-symmetric, Properties of Binary Relations, Closure, Partial Ordering Relations; Counting - Pigeonhole Principle.

Algebraic Structures: Algebraic Structures with one Binary Operation, Semi Groups, Monoids, Groups, Algebraic Structures with two Binary Operation, Rings, Integral Domain and Fields.

Permutation and Combination: Introduction to Permutation and Combination, Permutation of thing not all different, Multiplication Principle, Addition Principle.

Basics of probability: Random Experiment, sample space, event, types, definition, simple problems.

Mathematical Induction: Principle of Inclusion and Exclusion.

UNIT II. Growth of Functions: Asymptotic Notations, Summation Formulas and Properties, Bounding Summations, Approximation by Integrals.

UNIT III. Recurrences: Recurrence Relations, Generating Functions, Linear Recurrence Relations with Constant Coefficients and their Solution, Substitution Method, Recurrence Trees, Master Theorem.

UNIT IV. Graph Theory: Basic Terminology, Models and Types, Multigraphs and Weighted Graphs, Directed Graph, Graph Representation, Graph Isomorphism, Connectivity, Euler and Hamiltonian Paths and Circuits, Planar Graphs, Graph Coloring, Trees, Basic Terminology and Properties of Trees, Introduction to Spanning Trees.

UNIT V. Propositional Logic: Proposition or Statements, Truth table, Logical Connectives, Well-formed Formulas, Tautologies, Contradiction, Equivalences, Inference Theory, Conjunctive Normal Form, Disjunctive Normal Form.

References/ Suggested Readings:

1. C.L. Liu & Mahopatra, Elements of Discrete mathematics, 2nd Sub Edition 1985, Tata McGraw Hill.

2. Kenneth Rosen, Discrete Mathematics and Its Applications, Sixth Edition, McGraw Hill 2006

3. M. O. Albertson and J. P. Hutchinson, Discrete Mathematics with Algorithms 1988 John wiley Publication.

MAJOR COURSE - 4

Course Name: Digital Logic and Computer Organization

Course Code: BSCCOSMJ302

Course Type: **Major (Theoretical & Practical)**	Course Details: **MJC-4**			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 Structure of Computers: Basic units of Computer System; CPU – Control Unit, ALU; System Buses, Memory module Overview, Measuring unit of memory – Bit, Byte, KB, MB, GB, TB; Input Devices – Keyboard, Mouse, Scanner, Output Devices – Monitor, Printer, Basic Operational Concepts, Performance – Processor Clock, Clock Rate, Performance Measurement.

Unit II: Number System: Positional number systems; Representation of signed numbers- signed magnitude, one's complement, 2's complement representation techniques, Merits of 2's complement representation scheme; Various binary codes- BCD, excess -3, Gray code; Binary arithmetic- addition, subtraction, multiplication and division of unsigned binary numbers.

Unit III: Logic gates: Basic logic operations- logical sum(or), logical product (AND), complementation (not), Anti coincidence (EX-OR)and coincidence (EX-NOR) operations: Truth tables of Basic gates; Boolean Variables and Expressions; Demorgan's theorem; Universal gates- NAND and NOR; Boolean expressions Simplification- Algebraic technique, Karnaugh map technique, 3 variable and 4 variable Karnaugh map.

Unit IV: Combinational Circuits: Half adder, full adder, binary magnitude comparator, adder/subtractor circuits, multiplexer and demultiplexer circuits, BCD adder/subtractor; ALU; parity generators, code converters, priority encoders, PLAs.

Unit V: Sequential circuits: flip- flops, - RS, clocked RS, D, JK, T flip-flops,: Race condition, Master Slave JK: Registers, Universal Shift Registers; Counters- Binary, decade; modulo-r divider; Practical IC's; Sequential Machine design.

Unit VI. Machine Instructions and Programs: Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes, Assembly Language, Basic Input and Output Operations, Stacks and Queues, Subroutines, Additional Instructions, Encoding of Machine Instructions.

Unit VII: Input/Output Organization: Accessing I/O Devices, Interrupts – Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Controlling Device Requests, Exceptions, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces – PCI Bus, SCSI Bus, USB.

Unit VIII: Memory System: Basic Concepts, Semiconductor RAM Memories, Read Only Memories, Speed, Size, and Cost, Cache Memories – Mapping Functions, Replacement Algorithms, Performance Considerations, Virtual Memories, Secondary Storage.

Unit IX: Basic Processing Unit: Some Fundamental Concepts, Execution of a Complete Instruction, Multiple Bus Organization, Hard-wired Control, Micro-programmed Control.

Practical

Unit I: Study on the characteristic of AND, OR, NAND, NOR, EX-OR, EX-NOR gates.

Unit II: Design of different combinational circuit such as half adder/subtractor, full adder/subtractor, decoder/encoder, priority encoder, multiplexer, demultiplexer, magnitude comparator etc.

Unit III: Study on the characteristic of different flip-flops-JK, RS, T, D etc.

Unit IV: Design and implementation of different sequential circuit such as shift register, counter-decimal, ripple etc.

CA (Internal) Evaluation: Laboratory 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. M. Morris Mano, Digital Logic, Pearson.
2. D.P. Leach, A.P. Malvino and G. Saha, Digital Principles and Applications, McGraw-Hill.
3. Modern Digital Electronics: R.P. Jain, Tata McGraw Hill.
4. Computer Organization: Carl Hamacher, Zvonko Vranesic, Safwat Zaky, 5th Edition, Tata McGraw Hill, 2002.

MINOR COURSE - 3

Course Name: Digital Logic and Computer Organization

Course Code: BSCCOSMN301

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

Course Content:

Theory

Unit I: Basic Structure of Computers: Basic units of Computer System; CPU – Control Unit, ALU; System Buses, Memory module Overview, Measuring unit of memory – Bit, Byte, KB, MB, GB, TB; Input Devices – Keyboard, Mouse, Scanner, Output Devices – Monitor, Printer, Basic Operational Concepts, Performance – Processor Clock, Clock Rate, Performance Measurement.

Unit II: Number System: Positional number systems; Representation of signed numbers- signed magnitude, one's complement, 2's complement representation techniques, Merits of 2's complement representation scheme; Various binary codes- BCD, excess -3, Gray code; Binary arithmetic- addition, subtraction, multiplication and division of unsigned binary numbers.

Unit III: Logic gates: Basic logic operations- logical sum(or), logical product (AND), complementation (not), Anti coincidence (EX-OR)and coincidence (EX-NOR) operations: Truth tables of Basic gates; Boolean Variables and Expressions; Demorgan's theorem; Universal gates- NAND and NOR; Boolean expressions Simplification- Algebraic technique, Karnaugh map technique, 3 variable and 4 variable Karnaugh map.

Unit IV: Combinational Circuits: Half adder, full adder, binary magnitude comparator, adder/subtractor circuits, multiplexer and demultiplexer circuits, BCD adder/subtractor; ALU; parity generators, code converters, priority encoders, PLAs.

Unit V: Sequential circuits: flip- flops, - RS, clocked RS, D, JK, T flip-flops,: Race condition, Master Slave JK: Registers, Universal Shift Registers; Counters- Binary, decade; modulo-r divider; Practical IC's; Sequential Machine design.

Unit VI. Machine Instructions and Programs: Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes, Assembly Language, Basic Input and Output Operations, Stacks and Queues, Subroutines, Additional Instructions, Encoding of Machine Instructions.

Unit VII: Input/Output Organization: Accessing I/O Devices, Interrupts – Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Controlling Device Requests, Exceptions, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces – PCI Bus, SCSI Bus, USB.

Unit VIII: Memory System: Basic Concepts, Semiconductor RAM Memories, Read Only Memories, Speed, Size, and Cost, Cache Memories – Mapping Functions, Replacement Algorithms, Performance Considerations, Virtual Memories, Secondary Storage.

Unit IX: Basic Processing Unit: Some Fundamental Concepts, Execution of a Complete Instruction, Multiple Bus Organization, Hard-wired Control, Micro-programmed Control.

Practical

Unit I: Study on the characteristic of AND, OR, NAND, NOR, EX-OR, EX-NOR gates.

Unit II: Design of different combinational circuit such as half adder/subtractor, full adder/subtractor, decoder/encoder, priority encoder, multiplexer, demultiplexer, magnitude comparator etc.

Unit III: Study on the characteristic of different flip-flops-JK, RS, T, D etc.

Unit IV: Design and implementation of different sequential circuit such as shift register, counter-decimal, ripple etc.

CA (Internal) Evaluation: Laboratory 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. M. Morris Mano, Digital Logic, Pearson.
2. D.P. Leach, A.P. Malvino and G. Saha, Digital Principles and Applications, McGraw-Hill.
3. Modern Digital Electronics: R.P. Jain, Tata McGraw Hill.
4. Computer Organization: Carl Hamacher, Zvonko Vranesic, Safwat Zaky, 5th Edition, Tata McGraw Hill, 2002.

MULTIDISCIPLINARY COURSE - 3

Course Name: Impact of Artificial Intelligence in Education

Course Code: MDC313

Course Type: **MD (Theoretical)**	Course Details:**MD-3**			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: Overview of Artificial intelligence (AI) - Problems of AI, AI techniques, Agents & environment, nature of environment, Overview of the history of AI in Education, PEAS (Performance measure, Environment, Actuators, Sensors) Representation, Examples of PEAS with Educational Contexts. Overview of machine learning techniques commonly used in education applications.

Unit II. AI in Modern Education System: Why AI in Education?, Transforming Teaching Patterns: Personalized Learning, Smart Content Creation, Automated Administrative Operations, AI in Classroom for Adaptable Access, Curriculum Planning and Development, Self Directed Learning, Closing the skill gap. Examples of AI-based tools currently used in education to illustrate these transformations (e.g., adaptive learning platforms, AI teaching assistants), Intelligent Tutoring System, Assessment Automation, adaptive assessments and their role in personalized learning, teacher–student collaboration, Role of AI in teacher professional development and skill-building, AI-based analytics for tracking student performance.

Unit III. Applications: AI is natural language processing (NLP), Scope of NLP applications (e.g., essay grading, language translation for global classrooms), ChatGPT and other conversational AI's role in providing 24/7 student support and assistance.

Unit IV. EMIS and LMS: Data analytics in Education Management Information Systems (EMIS) and the evolution to Learning Management Systems (LMS), Case studies of colleges/universities using EMIS and LMS with AI integration for data-driven decision-making, AI-powered LMS that provide real-time insights and automation.

Unit V. Benefits: Global Access to Education, Enhanced Learning Outcomes, Time and Cost Efficiency, Scalability of AI-driven education for remote areas and underserved populations.

Unit VI. Challenges and Ethical Considerations: Data Privacy and Security, Bias and Discrimination, Plagiarism and Academic Integrity, Legal implications and data protection regulations (e.g., GDPR compliance in educational AI systems), AI transparency and explain ability to ensure ethical usage in educational systems.

Unit VII. Future Directions and Opportunities: Augmented and Virtual Reality, Lifelong Learning and Skill Development, AI Literacy and Ethics Education, Emerging trends like AI-driven micro-credentials and competency-based education, Role of AI in shaping future education policies

References/ Suggested Readings:

1. Miao, F.; Holmes, W.; Huang, R.; Zhang, H. *AI and Education: A Guidance for Policymakers*; UNESCO Publishing: Paris, France, 2021.
2. Luckin, R., Holmes, W., Griffiths, M. & Forcier, L.B. Intelligence Unleashed: an argument for Al in Education. London: Pearson, 2016
3. Montebello, M. (2017). AI injected e-learning: the future of Online Education. Berlín, Germany: Springer.
4. Dr. R. D. Padmavathy, Dr. Raju Talreja AI IN EDUCATION: TRANSFORMING LEARNING AND TEACHING, Redshine Publication, 2024.
5. Sadiku Matthew N O, Artificial Intelligence in Education, iUniverse.

Semester- IV

MAJOR COURSE - 5

Course Name: Operating System

Course Code: BSCCOSMJ401

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

Course Content:

Theory

UNIT I. Introduction to Operating System: What is Operating System? History and Evolution of OS, Basic OS functions, Resource Abstraction, Types of Operating Systems– Multiprogramming Systems, Batch Systems, Time Sharing Systems; Operating Systems for Personal Computers, Workstations and Hand-held Devices, Process Control & Real time Systems.

UNIT II. Operating System Organization and Process Characterization: Processor and User Modes, Kernels, System Calls and System Programs, System View of the Process and Resources, Process Abstraction, Process Hierarchy, Threads, Threading Issues, Thread Libraries; Process Scheduling, Non-Pre-emptive and Preemptive Scheduling Algorithms.

UNIT III. Process Management: Deadlock, Deadlock Characterization, Necessary and Sufficient Conditions for Deadlock, Deadlock Handling Approaches: Deadlock Prevention, Deadlock Avoidance and Deadlock Detection and Recovery.

UNIT IV. Inter Process Communication and Synchronization: Concurrent and Dependent Processes, Critical Section, Semaphores, Methods for Inter-process Communication; Process Synchronization, Classical Process Synchronization Problems: Producer-Consumer, Reader-Writer.

UNIT V. Memory Management: Physical and Virtual Address Space; Memory Allocation Strategies– Fixed and -Variable Partitions, Paging, Segmentation, Virtual Memory; Page Replacement Algorithms.

UNIT VI. File and I/O Management, Disk Scheduling, OS security: Directory Structure, File Operations, File Allocation Methods, Device Management, Pipes, Buffer, Shared Memory, Disk Scheduling (FCFS, SSTF, SCAN, C-SCAN, LOOK, C-LOOK), Security Policy Mechanism, Protection, Authentication and Internal Access Authorization.

UNIT VII. Android Operating System: Introduction to Android Operating System, Android Development Framework, Android Application Architecture, Android Process Management and File System.

Practical

UNIT I. Students are required to write and practically execute programs to solve following problems using C programming language.

1. WRITE A PROGRAM (using fork() and/or exec() commands) where parent and child execute: a) same program, same code. b) same program, different code. c) before terminating, the parent waits for the child to finish its task.
2. WRITE A PROGRAM to report behavior of Linux kernel including kernel version, CPU type and model. (CPU information)
3. WRITE A PROGRAM to report behavior of Linux kernel including information on configured memory, amount of free and used memory. (memory information)
4. WRITE A PROGRAM to print file details including owner access permissions, file access time, where file name is given as argument.
5. WRITE A PROGRAM to copy files using system calls.
6. Write programs to implement scheduling algorithms (FCFS, Round Robin, SJF, SRJF)
7. Write program to implement non-preemptive priority based scheduling algorithm.
8. Write program to implement preemptive priority based scheduling algorithm.
9. Write program to calculate sum of n numbers using thread library.
10. Write a program to implement first-fit, best-fit and worst-fit allocation strategies.

UNIT II. UNIX and Shell Scripts

1. External and internal commands of UNIX
2. What is shell and various type of shell, Various editors present in unix/linux
3. Different modes of operation in vi editor
4. What is shell script, Writing and executing the shell script
5. Shell variable (user defined and system variables)
6. System calls, Using system calls
7. Pipes and Filters
8. Decision making in Shell Scripts (If else, switch), Loops in shell
9. Functions
10. Utility programs (cut, paste, join, tr, uniq utilities), Pattern matching utility (grep).

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. A Silberschatz, P.B. Galvin, G. Gagne, Operating Systems Concepts, 8th Edition, John Wiley Publications 2008.
2. A.S. Tanenbaum, Modern Operating Systems, 3rd Edition, Pearson Education 2007.
3. Sumitabha, Das, Unix Concepts and Applications, Tata McGraw-Hill Education.
4. Nemeth Synder and Hein, Linux Administration Handbook, Pearson Education, 2nd Edition ,2010.
5. W. Richard Stevens, Bill Fenner, Andrew M. Rudoff, Unix Network Programming, The sockets Networking API, Vol. 1, 3rd Edition, 2014.
6. Yashavant Kanetkar , UNIX Shell Programming, BPB Publication.
7. Kernighan and Pike, The Unix Programming Environment, Prentice-Hall.

MAJOR COURSE - 6

Course Name: Object Oriented Design and Programming

Course Code: BSCCOSMJ402

Course Type: **Major (Theoretical & Practical)**	Course Details:**MJC-6**			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: **Principles of Object Oriented Programming (OOP)**: Software Evaluation, A Look at Procedure Oriented Programming, OOP Paradigm, Basic Concepts of OOP, Benefits of OPP, Application of OOP.

UNIT II: Introduction to C++: What is C++, A simple C++ Program, More C++ statements, Structure of C++ Program. Tokens, Expression and controls Structures, Keywords, Identifiers and Constants, C++ data types, Variables: Declaration, Dynamic initialization of variables, Reference variables, Operators in C++ : Scope resolution operator, Member deferencing Operators, Memory Management Operators, Manipulators, Type cast operators, Expressions and Control Structures. Functions The main() function, Function Prototyping, Call by reference, Return by reference, Inline function.

UNIT III: Classes and Objects: Introduction, Specifying a Class, Defining member Functions, Nesting of Member functions, Private member functions, Memory Allocation for Objects, Static Data members, Static Member Functions, Arrays within a Class, Arrays of Objects, Objects as Function Arguments, Friendly Functions, Returning Objects.

UNIT IV: Pointers: Declaration and initializing, Manipulation of pointers, Pointers to objects, this pointers, Arrays of Pointers to Objects.

UNIT V. Constructors and Destructors: Constructors, Parameterized Constructors, Multiple Constructors in a class, Copy constructor, Destructors.

UNIT VI. Polymorphisms: Function Overloading, Operator overloading, Overloading Unary Operators, Overloading Binary Operators, Type Conversions.

UNIT VII: Inheritance: Introduction, Defining Derived Classes, Single inheritance, Multiple inheritance, Hierarchical inheritance, Multilevel inheritance, Hybrid inheritance, Virtual Base Classes, Constructor in Derived Classes, Pointers to Derived Classes, Static and dynamic binding, Virtual Functions, Pure Virtual Functions.

UNIT VIII: **I/O Operations and Files**: C++ Stream Classes, Unformatted I/O Operations, Formatted I/O operations, Classes for File Streams, Opening and Closing a File : open() and close() functions, Manipulators of File Pointers, Sequential Input and output Operations, Error handling in File Operations.

UNIT IX. Exception Handling: Benefits of Exception Handling, Throwing an Exception, the Try Block, Catching an Exception, Exception Objects, Exception Specifications, Rethrowing an Exception, Uncaught Exceptions.

UNIT X. Templates: Class Templates and Function Templates, simple generic classes and generic function, simple example programs. Introduction to Standard Template Library (STL).

Practical

Students are required to understand the object-oriented concepts using C++. They are required to practice the concepts learnt in the theory. Some of the programs to be implemented are listed as follows:

1. Number of vowels and number of characters in a string.
2. Write a function called zeros maller() that is passed with two introduce arguments by reference and set the smaller of the number to zero. Write a main() program to access this function.
3. Demonstration of Class, Constructors, destructors, input and output functions, Objects
4. Demonstration of array of object.
5. Demonstration of friend functions.
6. Demonstration of operator overloading.
7. Demonstration of inheritance.
8. Using this pointer to return a value (return by reference).
9. Demonstration of virtual function.
10. Demonstration of static function.
11. Accessing a particular record in a student's file.
12. Demonstration of exception handling.
13. Demonstration of class template and function template

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. E.Balagurusamy, Object Oriented Programming through C++, TMH.
2. Lafore Robert, Object Oriented Programming in Turbo C++, Galgotia Publications.
3. Herbert Schildt, C++: The Complete Reference, McGraw Hill.
4. B. Stroutstrup, The C++ Programming Language, 3rd Edition, Pearson Education.
5. Ashok N Kamthane, Programming in C++, Pearson.

MINOR COURSE - 4

Course Name: Database Management System

Course Code: BSCCOSMN401

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

Course Content:

Theory

Unit I. Introduction: Basic Concept, Drawbacks of File Management; Advantages of DBMS; Layered Architecture of Database, Data Independence; Data Models; Schemas and Instances; Database Languages; Database Users, DBA; Data Dictionary; Functional Components of a DBMS.

Unit II. ER Model: Entity, Attributes and Relationship; Structural Constraints; Keys (candidate, super, foreign, primary); Weak & strong Entity Set; ER Diagram; Specialization and Generalization; Constraints of Specialization and Generalization; Aggregation.

Unit III. Relational Model: Basic Concepts of Relational Model; Relational Algebra, introduction to Tuple Relational Calculus.

Unit IV. SQL: Overview of Structured Query Language (SQL), DDL, DCL, DML commands, aggregate functions, create a database table, create relationships between database tables, modify and manage tables, queries, create view.

Unit V. Integrity Constraints: Domain Constraints, Referential Integrity.

Unit VI. Relational Database Design: Problems of Un-Normalized Database; Functional Dependencies, Derivation Rules, Closure of FD Set, Membership of a Dependency, Canonical Cover; Decomposition to 1NF, 2NF, 3NF or BCNF Using FDs; Lossless Join Decomposition & Dependency Preservation.

Unit VII. Transaction Processing: ACID properties, concurrency control.

Practical

Students are required to practice the concepts learnt in the theory by designing and querying a database for a chosen organization (Like Library, Transport etc). The teacher may devise appropriate weekly lab assignments to help students practice the designing, querying a

database in the context of example database. Some indicative list of experiments is given below.

Experiment 1: E-R Model Analyze the organization and identify the entities , attributes and relationships in it. . Identify the primary keys for all the entities. Identify the other keys like candidate keys, partial keys, if any.

Experiment 2: Concept design with E-R Model Relate the entities appropriately. Apply cardinalities for each relationship. Identify strong entities and weak entities (if any).

Experiment 3: Relational Model Represent all the entities (Strong, Weak) in tabular fashion. Represent relation ships in a tabular fashion.

Experiment 4: Normalization Apply the First, Second and Third Normalization levels on the database designed for the organization

Experiment 5: Practicing DDL commands, Creating databases, How to create tables, altering the database, dropping tables and databases if not required. Try truncate, rename commands etc.

Experiment 6: Practicing DML commands on the Database created for the example organization DML commands are used to for managing data within schema objects. Some examples: ● SELECT - retrieve data from the a database ● INSERT - insert data into a table ● UPDATE - updates existing data within a table ● DELETE - deletes all records from a table, the space for the records remain

Experiment 7: Querying practice queries (along with sub queries) involving ANY, ALL, IN, Exists, NOT EXISTS, UNION, INTERSECT, Constraints etc.

Experiment 8: Querying (continued…) Practice queries using Aggregate functions (COUNT, SUM, AVG, and MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.

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. R. Elmasri, S.B. Navathe, Fundamentals of Database Systems 6th Edition, Pearson Education,2010.
2. R. Ramakrishanan, J. Gehrke, Database Management Systems 3rd Edition, McGraw-Hill, 2002.
3. A. Silberschatz, H.F. Korth, S. Sudarshan, Database System Concepts 6th Edition, McGraw Hill, 2010.
4. R. Elmasri, S.B. Navathe Database Systems Models, Languages, Design and application Programming, 6th Edition, Pearson Education, 2013.
5. Ullman, Principles of Database Systems, Galgotia Publications.

SKILL ENHANCEMENT COURSE - 3

Course Name: Web Programming

Course Code: BSCCOSSE401

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

Course Content:

Practical

Students are directed to do a minor project based on the contents of the course below (UNIT I to UNIT VI) for internal and ESE evaluation.

UNIT I. Introduction to World Wide Web: Internet Standards, Introduction to WWW and WWW Architecture, Internet Protocols, Overview of HTTP, HTTP request – response, Generations of dynamic web pages.

UNIT II. User Interface Design: Introduction to HTML and HTML5, TML Tags, Formatting and Fonts, Commenting Code, Anchors, Backgrounds, Images, Hyperlinks, Lists, Tables, Frames, HTML Forms. The need for CSS, Introduction to CSS, Basic syntax and structure, Inline Styles, Embedding Style Sheets, Linking External Style, Backgrounds, Manipulating Text, Margins and Padding, Positioning using CSS.

UNIT III. Java Programming: Java Script, Introduction, Core features, Data types and Variables, Operators, Expressions, Functions, Objects, Array, Date and Math related Objects. JAVA Networking classes, TCP/IP Protocol Suite, File Transfer Protocol (FTP), Java Environment, Setup for Web Applications, JavaBean, Application Builder Tool, Bean Developer Kit (BDK), The Java Beans API, Introduction to EJB

UNIT IV. Database: Database basics, SQL, MySQL, PostgreSQL, JDBC API, Driver Types, Two-tier and Three-tier Models, Connection Overview, Transactions, Driver Manager Overview, Statement Overview, Result Set Overview, Types of Result Sets, Concurrency Types, Prepared Statement Overview

UNIT V. Java Applet and JSP: Java Web Programs and Applets, Web Application, Servlet, Servlet Life Cycle, Servlet Programming, Introduction to JSP, Life Cycle of a JSP Page, Translation and Compilation, Creating Static Content, Response and Page Encoding, Creating Dynamic Content, Using Objects within JSP Pages, JSP Programming

UNIT VI. Dot Net Framework: Introduction to Dot Net, Dot Net framework and its architecture, CLR, Assembly, Components of Assembly, DLL hell and Assembly Versioning, Overview to C#, Introduction to ASP.net, Asp.net Programming.

Internal (CA) Evaluation: Minor Project Report (15 marks), Demonstration of the minor project (10 marks), Viva-voce (5 marks).

ESE Evaluation: Presentation of the minor project (10 marks), Viva-voce (10 marks).

References/ Suggested Readings:

1. James Keogh, J2EE: The complete Reference.
2. John Brock, Arun Gupta, Geertjan Wielenga, Java EE and HTML5 Enterprise Application Development (Oracle Press)
3. James Holmes, Struts: The Complete Reference, 2nd Edition
4. Stephen Walther, Kevin Scott Hoffman, Nate Dudek, ASP.NET Unleashed
5. John Sharp, Microsoft Visual C# 2013 Step by Step.
6. A. Majumdar and P. Bhattacharyya, Database Management Systems, McGraw Hill Education.

VALUE ADDED COURSE - 2

Course Name: Digital and Technological Solutions

Course Code: VAC403

Course Type: **VA (Theoretical)**	Course Details:**VAC-2**			L-T-P: 4**–0–0**	
Credit: 4	Full Marks: **50**	CA Marks		ESE Marks	
		Practical	Theoretical	Practical	Theoretical
		--	**15**	**--**	**35**

Course Content:

Theory

UNIT I: **Introduction & Evolution of Digital Systems:** Role & Significance of Digital Technology. Information & Communication Technology & Tools. Computer System & its working, Software and its types. Operating Systems: Types and Functions. Problem Solving: Algorithms and Flowcharts. Communication Systems: Principles, Model & Transmission Media. Computer Networks & Internet: Concepts & Applications, WWW, Web Browsers, Search Engines, Messaging, Email, Social Networking. Computer Based Information System: Significance & Types. E-commerce & Digital Marketing: Basic Concepts, Benefits & Challenges.

UNIT II: Digital India & e-Governance: Initiatives, Infrastructure, Services and Empowerment. Digital Financial Tools: Unified Payment Interface, Aadhar Enabled Payment System, USSD, Credit / Debit Cards, e-Wallets, Internet Banking, NEFT/RTGS and IMPS, Online Bill Payments and PoS. Cyber Security: Threats, Significance, Challenges, Precautions, Safety Measures, & Tools, legal and ethical perspectives. Emerging Technologies & their applications: Overview of Cloud Computing, Big Data, Internet of Things, Virtual Reality, Blockchain & Cryptocurrency, Robotics, Machine Learning & Artificial Intelligence, 3-D Printing. Digital Signatures.

References/ Suggested Readings:

1. V. Rajaraman, Introduction to Information Technology, 3rd Edition, PHI;
2. E Balagurusamy, Fundamentals of Computers, Tata Mc GrawHill;
3. Behrouz A. Forouzan, Data Communications and Networking, McGraw Hill;
4. Pramod Kumar, Anuradha Tomar, R. Sharmila, Emerging Technologies in Computing Theory, Practice, and Advances, Edition 2021, Chapman and Hall/CRC Imprint;
5. Buvya, Broberg, and Gosciniski, Cloud Computing- Principals and Paradigms, Wiley
6. Russel and Norving, Artificial Intelligence- A Modern Approach, Pearson Education;
7. Samuel Greengard, Internet of Things, MIT Press;
8. C.S.V. Murthy, E-commerce Concepts, Models, Strategies;
9. Hurwith, Nugent Halper, Kaufman, Big Data for dummies, Wiley & Sons - Wiley.

Semester - V

MAJOR COURSE - 7

Course Name: Design and Analysis of Algorithms

Course Code: BSCCOSMJ501

Course Type: **Major (Theoretical)**	Course Details:**MJC-7**			L-T-P: **4–1–0**	
Credit: **5**	Full Marks: **100**	CA Marks		ESE Marks	
		Practical	Theoretical	Practical	Theoretical
		--	**30**	**--**	**70**

Course Content:

Theory

UNIT I. Introduction
Algorithm Design paradigms - motivation, concept of algorithmic efficiency, run time analysis of algorithms, Asymptotic Notations. Recurrences- substitution method, master method, Sliding Window Technique.

UNIT II. Divide and conquer
Structure of divide-and-conquer algorithms: examples; Binary search, quick sort, Merge sort, Strassen Multiplication; Analysis of divide and conquer run time recurrence relations.

UNIT III. Greedy Method
Overview of the greedy paradigm examples of exact optimization solution (minimum cost spanning tree), Approximate solution (Knapsack problem), Single source shortest paths, traveling salesman

UNIT IV. Dynamic programming
Overview, difference between dynamic programming and divide and conquer, Applications: Shortest path in graph, chain Matrix multiplication, Traveling salesman Problem, longest Common sequence, knapsack problem

UNIT V. Sorting and Searching Techniques
Elementary Sorting techniques– Bubble Sort, Insertion Sort, Merge Sort, Advanced Sorting techniques- Heap Sort, Quick Sort, Sorting in Linear Time - Bucket Sort, Radix Sort and Count Sort, Searching Techniques- Medians & Order Statistics, complexity analysis

UNIT VI. Graphs Algorithms
Graph Algorithms– Breadth First Search, Depth First Search and its Applications, Spanning tree, Minimum Spanning Trees (Kruskal and Prim's algorithms). String Processing (KMP Technique)

UNIT VII. Lower Bounding Techniques
Decision Trees, Balanced Trees (AVL, B tree, Red-Black Trees)

References/ Suggested Readings:

1. T.H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein Introduction to Algorithms, PHI, 3rd Edition 2009
2. Sara basse & A.V. Gelder, Computer Algorithm – Introduction to Design and Analysis, Publisher – Pearson 3rd Edition 1999
3. Knuth Donald E., The art of computer programming: Fundamental algorithms (Vol. 1), Pearson. 3rd Edition.
4. E. Horowitz, S. Sahni, and S. Rajsekaran, "Funadmentals of Computer Algorithms," Galgotia Publication.
5. Richard Bellman, "Dynamic Programming", Dover Publications Inc.

MAJOR COURSE - 8

Course Name: Database Management System

Course Code: BSCCOSMJ502

<table>
<tr><td>Course Type: Major (Theoretical & Practical)</td><td colspan="2">Course Details: MJC-8</td><td colspan="2">L-T-P: 3 – 0 – 4</td></tr>
<tr><td rowspan="3">Credit: 5</td><td rowspan="3">Full Marks: 100</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>30</td><td>15</td><td>20</td><td>35</td></tr>
</table>

Course Content:

Theory

Unit I. Introduction: Basic Concept, Drawbacks of File Management; Advantages of DBMS; Layered Architecture of Database, Data Independence; Data Models; Schemas and Instances; Database Languages; Database Users, DBA; Data Dictionary; Functional Components of a DBMS.

Unit II. ER Model: Entity, Attributes and Relationship; Structural Constraints; Keys (candidate, super, foreign, primary); Weak & strong Entity Set; ER Diagram; Specialization and Generalization; Constraints of Specialization and Generalization; Aggregation.

Unit III. Relational Model: Basic Concepts of Relational Model; Relational Algebra, introduction to Tuple Relational Calculus.

Unit IV. SQL: Overview of Structured Query Language (SQL), DDL, DCL, DML commands, aggregate functions, create a database table, create relationships between database tables, modify and manage tables, queries, create view.

Unit V. Integrity Constraints: Domain Constraints, Referential Integrity.

Unit VI. Relational Database Design: Problems of Un-Normalized Database; Functional Dependencies, Derivation Rules, Closure of FD Set, Membership of a Dependency, Canonical Cover; Decomposition to 1NF, 2NF, 3NF or BCNF Using FDs; Lossless Join Decomposition & Dependency Preservation.

Unit VII. Transaction Processing: ACID Properties, Transaction States, Locking Mechanisms: Shared locks, Exclusive locks, Deadlocks, Concurrency Control Techniques: Two-Phase Locking, Serializability and Recovery Mechanisms.

Practical

Students are required to practice the concepts learnt in the theory by designing and querying a database for a chosen organization (Like Library, Transport etc). The teacher may devise appropriate weekly lab assignments to help students practice the designing, querying a database in the context of example database.

Unit I: E-R Model Analyze the organization and identify the entities , attributes and relationships in it. . Identify the primary keys for all the entities. Identify the other keys like candidate keys, partial keys, if any.

Unit II: Concept design with E-R Model Relate the entities appropriately. Apply cardinalities for each relationship. Identify strong entities and weak entities (if any).

Unit III: Relational Model Represent all the entities (Strong, Weak) in tabular fashion. Represent relation ships in a tabular fashion.

Unit IV: Normalization Apply the First, Second and Third Normalization levels on the database designed for the organization.

Unit V: Practicing DDL commands, Creating databases, How to create tables, altering the database, dropping tables and databases if not required. Try truncate, rename commands etc.

Unit VI: Practicing DML commands on the Database created for the example organization DML commands are used to for managing data within schema objects. Some examples: ● SELECT - retrieve data from the a database ● INSERT - insert data into a table ● UPDATE - updates existing data within a table ● DELETE - deletes all records from a table, the space for the records remain

Unit VII: Querying practice queries (along with sub queries) involving ANY, ALL, IN, Exists, NOT EXISTS, UNION, INTERSECT, Constraints etc.

Unit VIII: Querying (continued…) Practice queries using Aggregate functions (COUNT, SUM, AVG, and MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.

Unit IX: Create triggers and test before/after Insert, Update and Delete events.

Internal (CA) Evaluation: Practical Note Book (15 marks), One experiment (10 marks) from Unit I to Unit IX, Viva-voce (5 marks)

ESE Evaluation: One experiment from Unit V to Unit IX (10 marks), Viva-voce (10 marks)

References/ Suggested Readings:

1. R. Elmasri, S.B. Navathe, Fundamentals of Database Systems 6th Edition, Pearson Education,2010.
2. R. Ramakrishanan, J. Gehrke, Database Management Systems 3rd Edition, McGraw-Hill, 2002.
3. A. Silberschatz, H.F. Korth, S. Sudarshan, Database System Concepts 6th Edition, McGraw Hill, 2010.
4. R. Elmasri, S.B. Navathe Database Systems Models, Languages, Design and application Programming, 6th Edition, Pearson Education, 2013.
5. Ullman, Principles of Database Systems, Galgotia Publications.

MAJOR COURSE - 9

Course Name: Computer Networks

Course Code: BSCCOSMJ503

Course Type: **Major (Theoretical & Practical)**	Course Details: **MJC-9**			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 Computer Networks and Networking Elements: Network Definition, Network Topologies, Network Classifications, Network Protocol, Layered Network Architecture, Overview of OSI Reference Model, Overview of TCP/IP Protocol Suite, Hub, Switch (Managed and Unmanaged), Routers.

UNIT II. Data Communication Fundamentals and Techniques: Analog and Digital Signal, Data-Rate Limits, Digital to Digital Line Encoding Schemes, Pulse Code Modulation, Parallel and Serial Transmission, Digital to Analog Modulation - Multiplexing Techniques-FDM, TDM, Transmission Media.

UNIT III. Networks Switching Techniques and Access Mechanisms: Circuit Switching, Packet Switching- Connectionless Datagram Switching, Connection Oriented Virtual Circuit Switching; Dial-Up Modems, Digital Subscriber Line, Cable TV for Data Transfer.

UNIT IV. Data Link Layer Functions and Protocol: Error Detection and Error Correction Techniques, Data-Link Control- Framing and Flow Control, Error Recovery Protocols-Stop and Wait ARQ, Go-Back-N ARQ, Point to Point Protocol on Internet.

UNIT V. Multiple Access Protocol and Network Layer: CSMA/CD Protocols, Ethernet LANS; Connecting LAN and Back-Bone Networks- Repeaters, Hubs, Switches, Bridges, Router and Gateways, Networks Layer Functions and Protocols, Routing, Routing Algorithms, Network Layer Protocol of Internet - IP Protocol, Internet Control Protocols.

UNIT VI. Transport Layer and Application Layer Functions and Protocols: Transport Services- Error and Flow Control, Connection Establishment and Release- Three Way Handshake, Overview of Application Layer Protocol, Overview of DNS Protocol; Overview of WWW & HTTP Protocol.

UNIT VII. Security and Cellular Communication: Firewall**,** Basics of cryptography; message security. Principles of secure communication, Evolution to 5G, VoLTE, 5G NR, 5G architecture.

Practical

UNIT I. Network Devices and Configuration

1. Identification of network devices like hub, switch, modem etc.
2. Use of ping and tracert / traceroute, ipconfig / ifconfig, route and arp utilities.
3. Configure LAN
4. Configure IP static routing.
5. Configure IP routing using RIP.

UNIT II. All programs should be developed in C/ C++ / Java / Python

1. Simulate Cyclic Redundancy Check (CRC) error detection algorithm for noisy channel.
2. Simulate Hamming-code based error detection & correction algorithm for noisy channel.
3. Simulate and implement stop and wait protocol for noisy channel.
4. Simulate and implement go back N sliding window protocol.
5. Simulate and implement selective repeat sliding window protocol.
6. Simulate and implement MST construction (Prim‟s, Kruskal‟s) for Ethernet
7. Simulate and implement the various routing algorithms (RIP, Distance-Vector routing, Dijkstra‟s, Bellman-Ford, Floyd-Warshall, Flooding)
8. Socket Programming.

Internal (CA) Evaluation: Practical Note Book (15 marks), One experiments from Unit I (10 marks), Viva-voce (5 marks)

ESE Evaluation: One experiment from Unit II (10 marks), Viva-voce (10 marks)

References/ Suggested Readings:

1. B. A. Forouzan: Data Communications and Networking, Fourth edition, THM Publishing Company Ltd 2007.
2. A. S. Tanenbaum: Computer Networks, Fifth edition, PHI Pvt. Ltd 2011
3. William Stallings:Data and Computer Communications, Eight Edition, Pearson.

MINOR COURSE - 5

Course Name: Operating System and Shell Scripts

Course Code: BSCCOSMN501

Course Type: **Minor (Theoretical & Practical)**	Course Details: **MNC-5**			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 Operating System: What is Operating System? History and Evolution of OS, Basic OS functions, Resource Abstraction, Types of Operating Systems– Multiprogramming Systems, Batch Systems, Time Sharing Systems; Operating Systems for Personal Computers, Workstations and Hand-held Devices, Process Control & Real time Systems.

UNIT II. Operating System Organization and Process Characterization: Processor and User Modes, Kernels, System Calls and System Programs, System View of the Process and Resources, Process Abstraction, Process Hierarchy, Threads, Threading Issues, Thread Libraries; Process Scheduling, Non-Pre-emptive and Preemptive Scheduling Algorithms.

UNIT III. Process Management: Deadlock, Deadlock Characterization, Necessary and Sufficient Conditions for Deadlock, Deadlock Handling Approaches: Deadlock Prevention, Deadlock Avoidance and Deadlock Detection and Recovery.

UNIT IV. Inter Process Communication and Synchronization: Concurrent and Dependent Processes, Critical Section, Semaphores, Methods for Inter-process Communication; Process Synchronization, Classical Process Synchronization Problems: Producer-Consumer, Reader-Writer.

UNIT V. Memory Management: Physical and Virtual Address Space; Memory Allocation Strategies– Fixed and -Variable Partitions, Paging, Segmentation, Virtual Memory; Page Replacement Algorithms.

UNIT VI. File and I/O Management, Disk Scheduling, OS security: Directory Structure, File Operations, File Allocation Methods, Device Management, Pipes, Buffer, Shared Memory, Disk Scheduling (FCFS, SSTF, SCAN, C-SCAN, LOOK, C-LOOK), Security Policy Mechanism, Protection, Authentication and Internal Access Authorization.

UNIT VII. Android Operating System: Introduction to Android Operating System, Android Development Framework, Android Application Architecture, Android Process Management and File System.

Practical

UNIT I. Students are required to write and practically execute programs to solve following problems using C programming language.

1. WRITE A PROGRAM (using fork() and/or exec() commands) where parent and child execute: a) same program, same code. b) same program, different code. c) before terminating, the parent waits for the child to finish its task.
2. WRITE A PROGRAM to report behavior of Linux kernel including kernel version, CPU type and model. (CPU information)
3. WRITE A PROGRAM to report behavior of Linux kernel including information on configured memory, amount of free and used memory. (memory information)
4. WRITE A PROGRAM to print file details including owner access permissions, file access time, where file name is given as argument.
5. WRITE A PROGRAM to copy files using system calls.
6. Write programs to implement scheduling algorithms (FCFS, Round Robin, SJF, SRJF)
7. Write program to implement non-preemptive priority based scheduling algorithm.
8. Write program to implement preemptive priority based scheduling algorithm.
9. Write program to calculate sum of n numbers using thread library.
10. Write a program to implement first-fit, best-fit and worst-fit allocation strategies.

UNIT II. UNIX and Shell Scripts

1. External and internal commands of UNIX
2. What is shell and various type of shell, Various editors present in unix/linux
3. Different modes of operation in vi editor
4. What is shell script, Writing and executing the shell script
5. Shell variable (user defined and system variables)
6. System calls, Using system calls
7. Pipes and Filters
8. Decision making in Shell Scripts (If else, switch), Loops in shell
9. Functions
10. Utility programs (cut, paste, join, tr, uniq utilities), Pattern matching utility (grep).

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. A Silberschatz, P.B. Galvin, G. Gagne, Operating Systems Concepts, 8th Edition, John Wiley Publications 2008.
2. A.S. Tanenbaum, Modern Operating Systems, 3rd Edition, Pearson Education 2007.
3. Sumitabha, Das, Unix Concepts and Applications, Tata McGraw-Hill Education.
4. Nemeth Synder and Hein, Linux Administration Handbook, Pearson Education, 2nd Edition ,2010.
5. W. Richard Stevens, Bill Fenner, Andrew M. Rudoff, Unix Network Programming, The sockets Networking API, Vol. 1, 3rd Edition, 2014.
6. Yashavant Kanetkar , UNIX Shell Programming, BPB Publication.
7. Kernighan and Pike, The Unix Programming Environment, Prentice-Hall.

Semester - VI

MAJOR COURSE - 10

Course Name: Cryptography and Network Security

Course Code: BSCCOSMJ601

Course Type: **Major (Theoretical)**	Course Details:**MJC-10**			L-T-P: **4–1–0**	
Credit: **5**	Full Marks: **100**	CA Marks		ESE Marks	
		Practical	Theoretical	Practical	Theoretical
		--	**30**	**--**	**70**

Course Content:

Theory

UNIT I. Introduction to Cyber Security: Cybersecurity objectives, Cybersecurity roles, Differences between Information Security & Cyber security. Confidentiality, integrity, & availability, Authentication & nonrepudiation, Types of attack.

UNIT II. Cryptography Concepts & Techniques: Introduction, Plaintext & Cipher text, Substitution Techniques, Transposition Techniques, Encryption & Decryption, Symmetric & Asymmetric key Cryptography, Key Range & Key Size.

UNIT III. Symmetric Key Algorithm: Introduction, Algorithm types & Modes, Overview of Symmetric Key Cryptography, DES(Data Encryption Standard) algorithm, IDEA (International Data Encryption Algorithm) algorithm, RC5 (Rivest Cipher 5) algorithm.

UNIT IV. Asymmetric Key Algorithm, Digital Signature and RSA: Introduction, Overview of Asymmetric key Cryptography, RSA algorithm, NP-hard, Symmetric & Asymmetric key Cryptography together, Digital Signature, Basic concepts of Message Digest and Hash Function (Algorithms on Message Digest and Hash function not required).

UNIT V. Firewall: Introduction, Types of firewall, Firewall Configurations, DMZ Network.

UNIT VI. Transport-level Security: Web security considerations, Secure Socket Layer and Transport Layer Security, HTTPS, Secure Shell (SSH)

Wireless Network Security: Wireless Security, Mobile Device Security, IEEE 802.11 Wireless LAN, IEEE 802.11i Wireless LAN Security

UNIT VII. Future Implications & Evolving Technologies: New & emerging IT & IS technologies, Mobile security issues, risks, & vulnerabilities, Cloud concepts around data & collaboration.

UNIT VIII. Electronic Mail Security: Pretty Good Privacy (PGP), S/MIME **IP Security:** IP Security overview, IP Security architecture, Authentication Header, Encapsulating security payload, Combining security associations, Internet Key Exchange

References/ Suggested Readings:

1. William Stallings, Cryptography and Network Security - Principles and Practice, Pearson Education, 6th Edition.
2. Atul Kahate, Cryptography and Network Security, Mc Graw Hill, 3rd Edition.
3. C. Kaufman, R. Perlman and M. Speciner, Network Security private communication in a public world, Pearson.
4. Wu Chwan-Hwa (John) et.al., Introduction To Computer Networks And Cybersecurity, BSP Books.
5. P. W. Singer and Allan Friedman, Cyber Security and Cyber War, Oxford University Press.

MAJOR COURSE - 11

Course Name: Theory of Computation

Course Code: BSCCOSMJ602

Course Type: **Major (Theoretical)**	Course Details:**MJC-11**			L-T-P: **4–1–0**	
Credit: **5**	Full Marks: **100**	CA Marks		ESE Marks	
		Practical	Theoretical	Practical	Theoretical
		--	**30**	**--**	**70**

Course Content:

Theory

UNIT I. Automata: Introduction to Formal Proof, Additional Forms of Proof, Inductive Proofs, Finite Automata (FA), Deterministic Finite Automata (DFA), Non-Deterministic Finite Automata (NFA), Finite Automata with Epsilon Transitions, Capabilities and Limitations of FA; State Equivalence and Minimization of FA, Melay and Moore Machines, Introduction to different types of languages and grammars.

UNIT II. Regular Expressions and Languages: Regular Expression, FA and Regular Expressions, Proving Languages not to be Regular (Pumping Lemma), Closure Properties of Regular Languages, Equivalence and Minimization of Automata.

UNIT III. Context Free Grammars and Languages: Context Free Grammar (CFG), Parse Trees , Ambiguity in Grammars and Languages, Definition of The Pushdown Automata, Languages of a Pushdown Automata, Equivalence of Pushdown Automata and CFG, Deterministic Pushdown Automata.

UNIT IV. Properties of Context Free Languages: Normal Forms for CFG, Pumping Lemma for CFL, Closure Properties of CFL,

UNIT V. Turing Machines: Turing Machine (TM) as a model of computation, Configuration of TM, Recursive and recursively enumerable languages, Variations of TM, Non Universal TM, Universal TM, decidability, Halting Problem.

UNIT VI. Undecidability: A Language that is not Recursively Enumerable (RE), an Undecidable Problem that is RE, Undecidable Problems about Turing Machine, Post's Correspondence Problem, The Classes P and NP.

References/ Suggested Readings:

1. J.E. Hopcroft, R. Motwani and J.D. Ullman, Introduction to Automata Theory, Languages and Computations, Pearson Education.
2. H.R. Lewis and C.H. Papadimitriou, Elements of the theory of Computation, Pearson Education.
3. K. L. P. Mishra and N. Chandrasekaran, Theory of Computer Science: Automata, Languages and Computation, PHI.
4. J. Martin, Introduction to Languages and the Theory of computation, Tata McGraw Hill.

MAJOR COURSE - 12

Course Name: Computer Graphics

Course Code: BSCCOSMJ603

Course Type: **Major (Theoretical & Practical)**	Course Details: **MJC-12**			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. Computer Graphics Basics: Basic elements of Computer graphics, Cathode Ray Tube, Raster Scan, Application of Computer Graphics. Architecture of Raster and Random scan display devices, input/output devices.

UNIT II. Output Primitives: Points and Lines, Line Generation Algorithm (DDA Algorithm, Bresenham's Line Generation, Mid-Point Algorithm), Line Function, Circle-Generating Algorithms (Bresenham's Algorithm and Midpoint Circle Algorithm), Properties of Circles, Ellipse-Generating Algorithm (Midpoint Ellipse Algorithm), Properties of Ellipses. Filled-Area Primitives, Scan-Line Polygon Fill Algorithm, Inside-Outside Tests, Scan-Line Fill of Curved Boundary, Areas Boundary-Fill Algorithm, Flood-Fill Algorithm, Fill-Area Functions.

UNIT III. Two-Dimensional Geometric Transformations: Basic Transformations (Translation, Rotation, Scaling), Matrix Representations and Homogeneous Coordinates, Composite Transformations (General Pivot-Point Rotation, General Fixed-Point Scaling, General Scaling Directions), Concatenation Properties, General Composite Transformations and Computational Efficiency, Other Transformations – Reflection, Shear, Transformations Between Coordinate Systems.

UNIT IV. Two-Dimensional Viewing: Viewing Coordinate Reference Frame, Window-to-viewport Coordinate Transformation, Clipping Operations, Point Clipping, Line Clipping (Cohen-Sutherland Line Clippings, Cyrus-Beck Line Clipping Algorithm), Polygon Clipping (Sutherland Hodgman Algorithm),Text Clipping, Curve Clipping, Exterior Clipping.

UNIT V. Three-Dimensional Geometric and Modeling Transformations: Translation, Rotation, Coordinate-Axes Rotations, General Three-Dimensional Rotations, Scaling, Reflections, Shears, Composite Transformations, Three-Dimensional Transformation Functions, Modeling and Coordinate Transformations.

UNIT VI. Hidden Surface Removal: Need for hidden surface removal, The Depth - Buffer Algorithm, Properties that help in reducing efforts, Scan Line coherence algorithm, Span – Coherence algorithm, Area-Coherence Algorithms, Warnock's Algorithm, Priority Algorithms

Practical

The students are required to create interactive graphics applications in C using graphics application programming interfaces and demonstrate geometrical transformations. The lab material includes implementation of line drawings, circle drawing, ellipse drawing, line clipping, polygon clipping, polygon filling as well as different geometrical transformations.

Experiment 1: Line Drawing Using DDA and Bresenham.

Experiment 2: Circle Drawing Using Midpoint Algorithm.

Experiment 3: Ellipse Drawing Using Midpoint Algorithm.

Experiment 4: Curve Generation: Bezier and B-Spline Curves.

Experiment 5: Line Clipping Algorithms- Cohen-Sutherland and Cyrus Beck.

Experiment 6: Sutherland–Hodgeman Polygon Clipping Algorithm.

Experiment 7: Polygon Filling Algorithms.

Experiment 8: Performing the basic 2D transformations such as translation, Scaling, Rotation, shearing and reflection for a given 2D object.

Internal (CA) Evaluation: Practical Note Book (15 marks), Two Experiments (10 marks) – one from Experiment 1 to 4 and another from Experiment 5 to 8, Viva-voce (5 marks).

ESE Evaluation: Two Experiments (10 marks) - one from Experiment 1 to 4 and another from Experiment 5 to 8, Viva-voce (10 marks).

References/ Suggested Readings:

1. Donald Hearn and M. Pauline Baker, Computer Graphics with Open GL, Prentice Hall.
2. R. K Maurya, Computer Graphics with Virtual Reality, Wiley.
3. Foley, Van Dam, Feiner and Hughes, Computer Graphics Principles & practice, Pearson Education.
4. D. P. Mukherjee, Fundamentals of Computer Graphics and Multimedia, PHI.

MAJOR COURSE - 13

Course Name: Software Engineering

Course Code: BSCCOSMJ604

Course Type: **Major (Theoretical & Practical)**	Course Details: **MJC-13**			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: Software engineering discipline – evolution and impact, Program vs S/W, Emergence of S/W engineering (Introduction to Control based design, Data structure-oriented design, data flow-oriented design, object-oriented design).

UNIT II: Software life cycle: Usefulness, Life cycle Model -Classical water fall model, Iterative waterfall model, prototype model, spiral model, comparative study of different models.

UNIT III: Software Requirement Specification: Role of system analyst, Need, Components and characteristic of SRS, Problems without a SRS, SRS document for Simple problems.

UNIT IV: Software Matrices: Halstead matrix, volume, size, difficulty, Effort estimation.

UNIT V: Software design: Cohesion & Coupling, S/W design Approach - Function oriented approach (DFD, Structure chart, Transformation of DFD into Structure chart), Object oriented approach (UML diagram, Use case model, class diagram, Interaction diagram)

UNIT VI: Coding: Coding standards, Code review - Code Walk through, Code Inspection, Clean room testing.

UNIT VII: Testing: Unit Testing (Driver and Stub Module, Black box testing [Equivalence class Partitioning and Boundary value analysis], White box testing [Statement coverage, Edge/branch coverage, condition coverage, path coverage, cyclomatic complexity]), Integration Testing (Big bang, Top down, Bottom up, Mixed approach), Verification and Validation of Software.

UNIT VIII: Maintenance: Characteristics, Types (corrective, adaptive and perfective), Software maintenance process model (Reverse engineering cycle followed by forward engineering model).

UNIT IX: Software Project Management: Introduction, Project planning, SPMP document; Metrics for project size estimation: Introduction, LOC, FP; Project Estimation Techniques: Introduction, Expert Judgment Technique, Software Metrics: LOC, KLOC, PM,

Delphi Cost Estimation, Introduction to the Rayleigh curve, COCOMO (Basic, Intermediate, Complete); Scheduling: Introduction, WBS, Activity Network, CPM, Gantt Charts, PERT Charts; Organization & Team Structure; Staffing; Risk Management; Software Configuration Management.

Practical

UNIT I. Development of problem statement.

UNIT II. Preparation of Software Requirement Specification Document, Design Documents and Testing Phase related documents.

UNIT III. Preparation of Software Configuration Management and Risk Management related documents.

UNIT IV. Study and usage of any Design phase CASE tool

UNIT V. Performing the Design by using any Design phase CASE tools.

UNIT VI. Develop test cases for unit testing and integration testing

UNIT VII. Develop test cases for various white box and black box testing techniques.

Students are directed to prepare a project report of the following sample projects (any one decides by the institution) considering above seven units:

1. Passport automation System
2. Online Exam Registration
3. Stock Maintenance System
4. Online course reservation system
5. E-ticketing
6. Credit Card Processing
7. E-book management System.
8. Recruitment system

Internal (CA) Evaluation: Project Report (20 marks), Viva-voce (10 marks).

ESE Evaluation: Presentation of the project (10 marks), Viva-voce (10 marks).

References/ Suggested Readings:

1. R. G. Pressman, Software Engineering, TMH
2. Sommerville, Ian, Software Engineering, Pearson Education
3. Pankaj Jalote, An Integrated Approach to Software Engineering, Narosa Publications.
4. Pfleeger, Shari Lawrence, Software Engineering Theory and Practice, Second Edition, Prentice- Hall 2001.
5. Object Oriented & Classical Software Engineering (Fifth Edition), SCHACH, TMH.

SUMMER INTERNSHIP- 1

Course Name: Summer Internship

Course Code: SI601

<table>
<tr><td>Course Type: SI
(Practical)</td><td colspan="2">Course Details: SI-1</td><td colspan="2">L-T-P: 0-0-4</td></tr>
<tr><td rowspan="3">Credit: 2</td><td rowspan="3">Full Marks: 50</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>30</td><td>-</td><td>20</td><td>-</td></tr>
</table>

Overview:
Computer Science proposes a Summer Internship Program keeping in view their feasible options/opportunities, to provide students with hands-on learning experiences, in alignment with the NCCF. NCCF emphasizes experiential learning, skill development, and multidisciplinary approaches, making internships a crucial component of higher education.

Duration:
Minimum 4 weeks programme (60 hours).

Mode:
Onsite training or Online training or blending (onsite and online both) mode

Key Areas of Internship:

- Artificial Intelligence (AI) and Machine Learning (ML)
- Data Science (Python, ML, AI)
- Bioinformatics
- Generative AI
- Cyber Security (Ethical Hacking)
- Internet of Things (IoT)
- Cloud Computing

Evaluation:

- On completion of the Summer Internship Programme, the students will submit a report with relevant photographs and an Authenticated Certificate jointly signed by the Supervisor/Mentor and the Head of the Institution.
- The report is to be signed by the Supervisor/Mentor with official seal.
- A viva-voce will be conducted by the Department with one Faculty acting as Internal Examiner and one External Examiner appointed from University
- The following Marks distribution is to be followed for evaluation
  CA: Submission of report: 20 marks, Demonstration: 10 marks.
  ESE: Presentation: 10 marks, Viva Voce: 10 marks

Collaboration:
The college is free to take initiative to approach any organization/HEI/research institute/centres/industry from the relevant domain of Computer Science. Apart from these, the College may consider other suitable/appropriate sectors for its students.