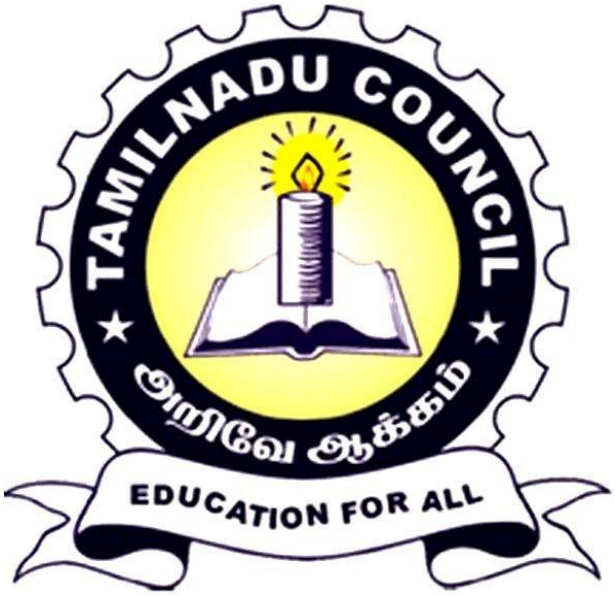

TAMILNADU COUNCIL FOR OPEN & DISTANCE LEARNING

INTEGRATED B.Sc.B.Ed.

(Computer Science)

SYLLABUS

TAMILNADU COUNCIL FOR OPEN AND DISTANCE LEARNING

Approved by International Council for Open & Distance Education (ICDE), Oslo, Norway

Internationally Accredited Institution Registered under Tamilnadu Govt Act

INTEGRATED B.Sc.B.Ed (Computer Science)

FIRST YEAR

S.No	SUBJECT CODE	SUBJECT	MAX.MARKS
1	IBED-CS1	English-I	100
2	IBED-CS2	Mathematics-I	100
3	IBED-CS3	Problem Solving	100
4	IBED-CS4	Fundamentals of Computing	100
5	IBED-CS5	Teacher in Emerging Indian Society	100
6	IBED-CS6	Development of Learner and Teaching -Learning Process	100
7	IBED-CS7	PRACTICAL-I	100

1.ENGLISH-I

UNIT-I Brief History of England

Tudor England- Stuart England -Restoration England -Revolutions - Eighteenth Century-19th Century Education- 20th Century

UNIT-II Literary Texts

R.K. Narayan- An Astrologer's Day and Sarojini Naidu - Bangle Sellers

UNIT-III Reading Comprehension

Definition of Comprehension- Types of Comprehension- Reading Materials- Vocabulary- Critical Reading- Effective Reading- Exercises

UNIT-IV Functional Grammars and Vocabulary

Parts of Speech- Tenses-Articles -Prepositions and Linkers –Punctuation- Common Mistakes -Polite Expression-Affixes

UNIT-V Language Skills

Reading Skills: SQ3R Technique -Writing Skills -Dictionary Use

2. MATHEMATICS-I

UNIT-I Properties of Matrices

Eigenvalues and Eigenvectors -Cayley Hamilton Theorem-Similar Matrices - Diagonalization of Matrices possessing Distinct Eigenvalues - Eigenvalues for symmetric matrices

UNIT-II Differential Calculus

Higher Derivatives – nth derivative – Standard Results – Trigonometric Transformations - Formation of Equations Involving Derivatives - Leibnitz's formula for nth derivative – Problems involving Leibnitz's formula - Methods of Integration of functions of the Following Types

$\frac{1}{(x+p)\sqrt{ax^2+bx+c}}$:$\sqrt{(x-a)(b-x)}$

$$\frac{1}{\sqrt{(x-a)(b-x)}}$$

$$\sqrt{\frac{(x-a)}{(b-x)}}$$

UNIT-III DIFFERENTIAL EQUATIONS

Partial Differential Equation-Formation of Equations by Elimination of Constants and an Arbitrary Function - Definition of General, Particular, Complete and Singular Integral - Solutions of First Order Equations in their Standard Forms - Lagrange's Method of Solving of Linear Equations

UNIT-IV FOURIER SERIES

Definition of Fourier Series - Finding Fourier Coefficients for a given Periodic Function with Period - Odd and Even Functions - Half – Range Series - Development in sine and cosine series

UNIT-V LINEAR PROGRAMMING PROBLEM

Formulation of LPP – Graphical Method – Simplex Method

3.PROBLEM SOLVING

UNIT-I

Invariants – Chocolate-bar Problem – Empty-box Problem – Tumbler Problem – Tetrominoes – Goat-cabbage-wolf Problem – Brute-force Search – State-space Explosion – Abstraction – Nervous-couples Problem – Denoting States and Transitions – Rule of Sequential Composition – Bridge Problem – Conditional Statements.

UNIT-II

Games – Matchstick Game – Winning Strategies – Subtraction-set Games – Sums of Games – Knights and Knaves – Logic Puzzles – Calculational Logic – Equivalence and Continued Equalities – Negation.

UNIT-III

Induction – Black and White Colouring – Cutting the Plane – Triomino Problem – Looking for Patterns – The Need for Proof – From Verification to Construction.

UNIT-IV

Fake-Coin Detection – Problem Formulation – Problem Solution – Tower of Hanoi Problem – Specification and Solution –Inductive Solution – Iterative Solution –Principles of Algorithm Design – Iteration. Invariants and Making Progress – Sam Loyd's Chicken-Chasing Problem.

UNIT-V

Bridge Problem – Lower and Upper Bounds – Outline Strategy – Regular Sequences – Sequencing Forward Trips – Choosing Settlers and Nomads – Knight's Circuit – Straight-Move Circuits – Super Squares – Partitioning the Board.

4.FUNDAMENTALS OF COMPUTING

UNIT-I Introduction to Computers

Introduction, Definition, Characteristics of computer, Evolution of Computer, Block Diagram Of a computer, Generations of Computer, Classification Of Computers, Applications of Computer, Capabilities and limitations of computer.

UNIT-II Basic Computer Organization:

Block Role of I/O devices in a computer system. Input Units: Keyboard, Terminals and its types. Pointing Devices, Scanners and its types, Voice Recognition Systems, Vision Input System, Touch Screen, Output Units: Monitors and its types. Printers: Impact Printers and its types. Non-Impact Printers and its types, Plotters, types of plotters, Sound cards, Speakers.

UNIT-III Storage Fundamentals:

Block Primary Vs Secondary Storage, Data storage & retrieval methods. Primary Storage: RAM ROM, PROM, EPROM, EEPROM. Secondary Storage: Magnetic Tapes, Magnetic Disks. Cartridge tape, hard disks, Floppy disks Optical Disks, Compact Disks, Zip Drive, Flash Drives.

UNIT-IV Software:

Block Software and its needs, Types of S/W. System Software: Operating System, Utility Programs Programming Language: Machine Language, Assembly Language, High Level Language their advantages & disadvantages. Application S/W and its types: Word Processing, Spread Sheets Presentation, Graphics, DBMS s/w. Operating System: Functions, Measuring System Performance, Assemblers, Compilers and Interpreters. Batch Processing, Multiprogramming, Multi-Tasking, Multiprocessing, Time Sharing, DOS, Windows, Unix/Linux.

UNIT-V Data Communication:

Communication Process, Data Transmission speed, Communication Types (modes), Data Transmission Medias, Modem and its working, characteristics, Types of Networks, LAN Topologies, Computer Protocols, Concepts relating to networking.

5. TEACHER IN EMERGING INDIAN SOCIETY

UNIT–I

Education: nature and meaning-its objectives in relation to the time and place. Education in the western context: with specific reference to Rousseau. Pestalozzi, Dewey, Russell, their impact on educational thought and class room practices, in term of progress trends in education. Indian thought and its contribution to educational practices.

UNIT–II

Philosophy and education: significance of studying philosophy in understanding educational practices and problem. Major philosophical systems-their salient features and their impact on education.

- Realism with reference to Aristotle and Jainism.
- Naturalism with reference to the views of Rousseau and Rabindranath Tagore.
- Idealism with reference to Plato, Socrates and Advaita Philosophy.
- Pragmatism with reference to Dewey “Instrumentalism and Experimentalism.
- Humanism-Historical & Scientific and Buddhists.

UNIT–III

Educational thinkers and their contribution in developing principles education.

- M.K. Gandhi: Basic tenets of Basic education.
- Swami Vivekananda: Man making education.
- Sri Aurobindo: Integral education, its basic premises; stages of development.
- Froebel: The play way method.
- Montessori: The didactic apparatus.

UNIT–IV

Knowledge about the directive principles in Indian Constitution; various articles mentioned in the constitution that are related to education; meaning of secularism, social goals; democracy and socialistic pattern of society; meaning of term "National integration and Emotional integration", factors contributing for achieve them.

UNIT–V

Sociological basis of education, education as an agent of social change, education as a means of National welfare through the immediate welfare of the society, education and human resource development.

UNIT–VI

Meaning of National integration and its need, role of teacher and educational institution in achieving National Integration through democratic interaction, explanation of cultural heritage, contributions of different religions (Hinduism, Buddhism, Sikhism, Islam, Christianity and Jainism) for the same cause and human upliftment, equal communication, philosophy of celebration of Indian festivals.

UNIT–VII

Meaning of new social order, eradication of illiteracy, objectives of NAEP; provisions made and channels started for educating socially, culturally and economically deprived, State's new programmes and Nations Programmes like NCC, NSS, etc.

6. DEVELOPMENT OF LEARNER AND TEACHING - LEARNING PROCESS

UNIT–I

Nature of psychology and learners

- Psychology: Its meaning, nature, methods and scope, functions of educational psychology.
- Stages of human development; stage specific characteristics and development tasks.
- Adolescence in Indian context-characteristics and problems of adolescents; their needs and aspirations.
- Guiding and counselling adolescents.

UNIT–II

Learning and motivation

- Nature of Learning; learning theories-Behaviourists (Skinner's),

Pavlov's Classical conditioning, Thorndike's connectionism and

Kohler's insight learning theory.

- Factors influencing learning and teaching process; learner related; teacher related; process related and content related.
- Motivation-nature, types; techniques of enhancing learner's

motivation.

UNIT–III

Intelligence

- Nature and characteristics of intelligence and its development.
- Theories of intelligence, two factor theory-Multifactor Theory (PMA) and SI Model.
- Measuring intelligence-Verbal, non-verbal and Performance test (one
- representative of group test and individual test of each).

UNIT–IV

Exceptional Children

- Concept of exceptional children's types, and characteristics of each type including children with learning disabilities.
- Individual differences-nature; accommodating individual differences in classroom.
- Learner centred techniques for teaching exceptional children.

UNIT–V

Personality

Definition, meaning-nature; development of personality; type & theories of personality.

UNIT–VI

Educational Statistics

- Data, Frequency distribution and tabulations.
- Calculation and uses-Central tendencies (Mean and mode) and variability with special reference to standard deviation.
- Correlation (Rank difference; meaning uses and calculation).

7. PRACTICAL-I

Exercises Based on

1. Introduction to MS office
2. Word Processing: Formatting Text, Pages, Lists, Tables
3. Spreadsheets: Worksheets, formatting data, creating charts and graphs, using formulas and functions, macros, Pivot Table.
4. Presentation Tools: Adding and formatting text, pictures, graphic objects, including charts, objects, formatting slides, notes, hand-outs, slide shows, using transitions, animations

TAMILNADU COUNCIL FOR OPEN AND DISTANCE LEARNING

Approved by International Council for Open & Distance Education (ICDE), Oslo, Norway

Internationally Accredited Institution Registered under Tamilnadu Govt Act

INTEGRATED B.Sc.B.Ed. (Computer Science)

SECOND YEAR

S.No	SUBJECT CODE	SUBJECT	MAX.MARKS
1	IBED-CS8	English-II	100
2	IBED-CS9	Mathematics-II	100
3	IBED-CS10	C Programming	100
4	IBED-CS11	Digital Electronics	100
5	IBED-CS12	Development Of Educational System in India	100
6	IBED-CS13	Guidance And Counselling	100
7	IBED-CS14	Practical-II	100

1.ENGLISH-II

Unit-I: Introduction to Effective Communication

- Importance of communication skills in B.Sc. studies and careers.
- The seven Cs of effective communication (Clarity, Conciseness, Correctness, Completeness, Courtesy, Concreteness, and Consideration)
- Active listening skills

Unit-II: Reading Comprehension & Analysis

- Techniques for effective reading (scanning, skimming, etc.)
- Reading comprehension passages from various academic disciplines relevant to science.
- Analysing and interpreting factual and opinion-based texts.
- Identifying key points, arguments, and supporting evidence.

Unit-III: **Writing Skills**

- Paragraph writing (different structures, topic development)
- Essay writing (argumentative, analytical, descriptive)
- Report writing (scientific reports, lab reports)
- Technical writing (instructions, summaries)
- Business communication (emails, proposals)

Unit- IV: Grammar & Vocabulary Development

- Review of essential grammar rules (tenses, subject-verb agreement, sentence structure)
- Enhancing vocabulary through prefixes, suffixes, and root words
- Academic and technical vocabulary specific to science fields

Unit-V: Communication Skills in the Workplace (Presentations, Job application letters, Group discussions)

- Literature & Science (Short stories or essays with a scientific theme)
- Soft Skills Development (Teamwork, Time Management, Problem-solving)

2.MATHEMATICS-II

Unit-I: **Sets, Relations & Functions**

- Set theory: Basic operations on sets (union, intersection, difference, complement), Venn diagrams.
- Relations: Types of relations (binary, equivalence, order), Functions: types of functions (one-one, many-one, onto), Composition of functions.

Unit-II: **Logic & Propositional Calculus**

- Propositional logic: Statements, connectives (AND, OR, NOT, etc.), Truth tables, Tautologies, Contradictions.
- Applications of logic in computer science (program flow, Boolean expressions)

Unit-III: **Linear Algebra**

- Matrices: Operations on matrices (addition, subtraction, transpose, multiplication), Inverse of a matrix, Solving systems of linear equations using matrices.
- Eigenvalues and Eigenvectors (basic concepts and applications)

Unit -IV :

- **Discrete Mathematics** (Graph Theory, Counting Techniques)
- **Probability & Statistics** (Basic concepts, Probability distributions)
- **Numerical Methods** (Solution of equations, Interpolation)

3.C PROGRAMMING

Unit-I: Introduction to Programming & C Language Basics

- Introduction to computers, programming concepts (algorithms, data types, variables)
- C language overview, basic structure of a C program (keywords, identifiers, statements)
- Input/Output operations in C (printf, scanf)

Unit-II: Operators & Control Flow

(Types of operators (arithmetic, relational, logical, assignment)

- Control flow statements (if-else, switch, loops - for, while, do-while)
- Nesting of control flow statements

Unit-III: Functions

- Function definition, function calls, parameter passing (by value, by reference)
- Recursion concept and applications
- Modular programming using functions

Unit-IV: Arrays

- One-dimensional and Multi-dimensional arrays (declaration, initialization, accessing elements)
- Array operations (traversing, searching, sorting)
- Passing arrays to functions

Unit -V: Pointers, Structures, Preprocessor Directives and Files

- Pointers (Concept of pointers, pointer arithmetic, memory allocation)
- Structures & Unions (Defining structures, accessing members, unions)
- Introduction to Preprocessor Directives (#define, #include, etc.)
- File Handling (Opening, closing, reading, writing files)

4.DIGITAL ELECTRONICS

Unit-I: Introduction to Digital Systems

- Review of number systems (binary, octal, hexadecimal) and their conversions
- Boolean algebra: basic operations (AND, OR, NOT, XOR, etc.), De Morgan's Laws
- Logic gates (AND, OR, NOT, NAND, NOR, XOR, XNOR) - Truth tables, symbol representations

Unit-II: Combinational Logic Circuits

- Simplification techniques (K-Map, Boolean expressions)
- Combinational circuit design (adders, subtractors, comparators, multiplexers, decoders)
- Digital Arithmetic (binary addition, subtraction, multiplication, division)

Unit-III: Sequential Logic Circuits

- Introduction to flip-flops (RS, JK, D, T) - operation, timing diagrams
- Registers and counters (synchronous and asynchronous)
- Finite State Machines (FSM) - concepts, state diagrams

Unit-IV: Digital Integrated Circuits

- Introduction to Integrated Circuits (ICs) - classification (SSI, MSI, LSI, VLSI)
- Digital logic families (TTL, CMOS) - characteristics, comparison
- Memory devices (RAM, ROM, PROM, EPROM) - types, operation

Unit-V: Troubleshooting and Design Tools

- Digital Troubleshooting Techniques (fault identification, testing methods)
- Introduction to Programmable Logic Devices (PLDs) (CPLDs, FPGAs)
- Computer Aided Design (CAD) for Digital Circuits (basic concepts, design tools)

5.DEVELOPMENT OF EDUCATIONAL SYSTEM IN INDIA

UNIT –I Educational Statistics

- Data, Frequency distribution and tabulations.
- Calculation and uses-Central tendencies (Mean and mode) and variability with special reference to standard deviation.
- Correlation (Rank difference; meaning uses and calculation). Education in India during (a) Vedic, (b) Buddhist and (c) Medieval periods.

UNIT –II

- Macaulay's minutes and Bentinck's resolution of 1835, Adam's report and its recommendations.
- Wood's Dispatch -1854.
- Lord Curzon's educational policy, Growth of national consciousness, National education movement.

UNIT –III

- Recommendations of Indian Education Commission-1882 its influence on the subsequent
- development of education.
- Essential features of Sadler Commission Report- 1917.
- Wardha scheme of education- 1937.

UNIT –IV

- University Education Commission (1948-49).
- Secondary Education Commission (1952-53).
- Indian Education Commission (1964-66).
- National Policy of Education (1986).
- Revised National Policy 1986 with modification made 1992.

UNIT –V

- Development of teacher education in India-Objectives
- Problems of teacher education in India.
- Role and functions of N.C.T.E.

UNIT –VI

- 6Primary education and its major problems (Universalization wastage and stagnation).
- Secondary education and its major problems (Expansion and vocationalisation).
- Higher education and its major problems (Autonomy and privatization).

6. GUIDANCE AND COUNSELLING

UNIT –I Understanding Guidance and counselling

- Guidance: Concept, aims, objectives, functions and principles. Role of guidance in growing up.
- Difference between guidance and counselling
- Ethical consideration while imparting these programs
- Technique and procedures of counselling
- Need & Procedure for (Educational, Psychological and Social) guidance.
- Group Guidance: Concept, Need, Significance and Principles, Organization of Guidance programs in schools. Major issues.

UNIT –II

- Macaulay's minutes and Bentinck's resolution of 1835,
- Adam's report and its recommendations.
- Wood's Dispatch -1854.
- Lord Curzon's educational policy, Growth of national consciousness, National education movement.

UNIT –III

- Approaches of Counselling
- Counselling: Meaning, Principles and approaches of counselling, Individual and Group Counselling.
- Process of counselling (Initial disclosure, In-depth Exploration & Commitment to Action) identification of problems.
- Qualities (including Skills for Listening, Questioning, Responding, Communicating) & Qualifications of a good / effective Counsellor
- Role of teacher as a Counsellor, Professional Ethics and Code of Conduct Different approaches to counselling
- Special counselling population, challenges of multi-cultural counselling. Techniques and Procedures of Guidance
- Need and procedure of guidance in educational, psychological and social difficulties;
- Assessment in guidance: Standardized (viz. Aptitude, attitude, interest, achievement, personality) and non-standardized tests (viz. questionnaire, observation, scale, anecdotal record, case study, interviews) technique.

UNIT –IV

- Occupational Information and Guiding Students with Special Needs
- Meaning, collection, types, classification & dissemination of occupational information, Career development: Teacher's role in career planning. Vocational training and placement opportunities for CWSN.
- Behaviour problems of students with special needs, viz. socio-emotional problems of children with disabilities and deprived groups such as SC, ST and girls, provision of facilities at governmental and non-governmental level and their utilization. Behaviour modification techniques
- Ethical and legal guidelines
- Dealing with depression and academic stress (with regard to their identification and intervention)

7.PRACTICAL-II

Unit I – Basic Programs

- Introduction to C programming environment, compilation, and execution.
- **Experiments / Sample Programs:**
    1. Program to display "Hello World".
    2. Program to declare and print variables of different data types.
    3. Program to add two numbers.
    4. Program to swap two numbers (with and without temporary variable).

5. Program to convert temperature from Celsius to Fahrenheit and vice versa.

Unit II – Decision Making Constructs

- Conditional statements (if, if-else, nested if, switch-case).
- **Experiments / Sample Programs:**
    1. Program to check whether a number is even or odd.
    2. Program to find the largest of three numbers.
    3. Program to check whether a year is a leap year.
    4. Program to calculate grade of a student based on marks.
    5. Program to create a simple calculator using switch-case.

Unit III – Looping Constructs

- Loops (for, while, do-while), nested loops.
- **Experiments / Sample Programs:**
    1. Program to print the first *n* natural numbers.
    2. Program to find factorial of a number.
    3. Program to generate Fibonacci series.
    4. Program to check whether a number is prime.
    5. Program to reverse a number.
    6. Program to find sum of digits of a number.
    7. Program to print patterns using nested loops (stars, numbers).

Unit IV – Arrays and Strings

- One-dimensional and two-dimensional arrays, string operations.
- **Experiments / Sample Programs:**
    1. Program to find the largest and smallest element in an array.
    2. Program to perform linear search and binary search.
    3. Program to sort elements of an array (bubble/selection sort).
    4. Program to perform matrix addition and subtraction.
    5. Program to perform matrix multiplication.
    6. Program to find the transpose of a matrix.
    7. Program to check whether a string is a palindrome.

8. Program to count vowels, consonants, digits, and spaces in a string.

Unit V – Functions and Recursion

- User-defined functions, recursion, variable scope.
- **Experiments / Sample Programs:**
    1. Program to find factorial using recursion.
    2. Program to generate Fibonacci series using recursion.
    3. Program to calculate GCD and LCM using functions.
    4. Program to demonstrate local and global variables.
    5. Program to illustrate static variable usage.

Unit VI – Pointers and Dynamic Memory

- Pointers, pointer arithmetic, pointers with arrays and functions, memory allocation.
- **Experiments / Sample Programs:**
    1. Program to swap two numbers using pointers.
    2. Program to demonstrate pointer to an array.
    3. Program to find the sum of elements using pointers.
    4. Program to allocate memory dynamically using malloc () and calloc ().
    5. Program to resize memory using realloc () and free it using free ().

Unit VII – Structures and Unions

- Structures, arrays of structures, nested structures, unions.
- **Experiments / Sample Programs:**
    1. Program to store and display student details using structures.
    2. Program to add two distances (feet–inches) using structures.
    3. Program using array of structures (e.g., book details).
    4. Program to demonstrate use of union and compare with structure.

Unit VIII – File Handling

- File operations: reading, writing, appending.
- **Experiments / Sample Programs:**
    1. Program to write and read text from a file.
    2. Program to count number of lines, words, and characters in a file.
    3. Program to append data to an existing file.
    4. Program to copy contents of one file into another.

TAMILNADU COUNCIL FOR OPEN AND DISTANCE LEARNING

Approved by International Council for Open & Distance Education (ICDE), Oslo, Norway

Internationally Accredited Institution Registered under Tamilnadu Govt Act

INTEGRATED B.Sc.B.Ed. (Computer Science)

THIRD YEAR

S.No	SUBJECT CODE	SUBJECT	MAX.MARKS
1	IBED-CS15	Principles of Programming Languages	100
2	IBED-CS16	Programming using C++	100
3	IBED-CS17	Data Structures	100
4	IBED-CS18	Statistics	100
5	IBED-CS19	Educational Psychology	100
6	IBED-CS20	Environmental Education	100
7	IBED-CS21	Practical-III	100

1.Principles of Programming Languages

Unit-I: Introduction to Programming Languages

- Definition and goals of programming languages
- Historical evolution of programming languages
- Classification of programming languages (imperative, declarative, functional, object-oriented, etc.)
- Syntax, semantics, and pragmatics of programming languages

Unit-II: Lexical Analysis and Parsing

- Lexical analysis: regular expressions, finite automata
- Parsing: context-free grammars, parsing techniques (top-down, bottom-up)
- Syntax trees and abstract syntax trees

Unit-III: Type Systems and Type Checking

- Types and type systems
- Static and dynamic type checking
- Type inference
- Polymorphism and type polymorphism

Unit-IV: Control Flow and Data Structures

- Control flow statements (if-else, loops, switch)
- Functions and procedures
- Data structures (arrays, records, lists, trees, graphs)
- Memory management and garbage collection

Unit -V: Object-Oriented Programming

- Concepts of objects, classes, and inheritance
- Polymorphism and dynamic binding
- Encapsulation and abstraction
- Design patterns

Unit-VI: Functional Programming

- Lambda expressions and higher-order functions
- Recursion and tail recursion
- Immutable data structures
- Lazy evaluation

Unit -VII: Concurrency and Parallelism

- Threads and processes
- Synchronization mechanisms (locks, semaphores, monitors)
- Parallel programming paradigms (shared memory, message passing)

Unit-VIII: Programming Language Design and Implementation

- Language design principles (readability, writability, reliability, efficiency)
- Language implementation techniques (interpreters, compilers)
- Virtual machines and bytecode

Unit-IX: Advanced Topics

- Domain-specific languages (DSLs)
- Metaprogramming and reflection
- Programming language theory (formal semantics, type theory)

2.PROGRAMMING USING C++

Unit-I: Introduction to C++

- History and evolution of C++
- C++ Standard Library
- Basic syntax and structure of C++ programs

Unit-II: Data Types and Variables

- Built-in data types (int, float, char, bool)

- User-defined data types (enum, struct, union)
- Variables and constants

Unit-III: Operators and Expressions

- Arithmetic, relational, logical, and bitwise operators
- Operator precedence and associativity
- Expressions and their evaluation

Unit-IV: Control Flow Statements

- Conditional statements (if-else, switch)
- Looping statements (for, while, do-while)
- Break and continue statements

Unit-V: Functions

- Function definition, declaration, and call
- Function overloading and default arguments
- Recursion

Unit-VI: Arrays and Pointers

- One-dimensional and multi-dimensional arrays
- Pointer declaration, dereferencing, and arithmetic
- Dynamic memory allocation and deallocation

Unit-VII: Classes and Objects

- Object-oriented programming concepts
- Class definition, member variables, and member functions
- Object creation and destruction
- Encapsulation and information hiding

Unit-VIII: Inheritance and Polymorphism

- Inheritance types (single, multiple, hierarchical, hybrid)
- Polymorphism (compile-time and runtime)
- Virtual functions and the virtual table

Unit-IX: Templates

- Function templates and class templates
- Template specialization

Unit-X: Exception Handling

- Try-catch-throw blocks
- Exception classes and user-defined exceptions

Unit-XI: Standard Template Library (STL)

- Containers (vector, list, map, set)
- Iterators
- Algorithms

3. Data Structures

Unit-I: Introduction to Data Structures

- Definition and importance of data structures
- Abstract data types (ADTs)
- Time and space complexity analysis

Unit-II: Arrays

- One-dimensional and multi-dimensional arrays
- Array operations (insertion, deletion, searching)
- Applications of arrays

Unit-III: Linked Lists

- Singly linked lists
- Doubly linked lists
- Circular linked lists
- Operations (insertion, deletion, traversal)
- Applications of linked lists

Unit-IV: Stacks and Queues

- Stack ADT
- Implementation using arrays and linked lists
- Applications of stacks (function calls, expression evaluation)
- Queue ADT
- Implementation using arrays and linked lists
- Applications of queues (breadth-first search, scheduling)

Unit-V: Trees

- Binary trees
- Binary search trees
- AVL trees
- Red-black trees
- Tree traversal algorithms (preorder, inorder, postorder)
- Applications of trees (sorting, searching, decision making)

Unit-VI: Heaps

- Min-heaps and max-heaps
- Heap operations (insertion, deletion, heapify)
- Applications of heaps (priority queues, sorting)

Unit-VII: Graphs

- Graph representations (adjacency matrix, adjacency list)
- Graph traversal algorithms (depth-first search, breadth-first search)
- Shortest path algorithms (Dijkstra's algorithm, Bellman-Ford algorithm)
- Minimum spanning trees (Kruskal's algorithm, Prim's algorithm)

Unit-VIII: Hashing

- Hash functions
- Collision resolution techniques (separate chaining, open addressing)
- Applications of hashing (set operations, database indexing)

Unit-IX: Advanced Data Structures

- Tries
- Disjoint sets
- B-trees and B+ trees
- Suffix trees

4. STATISTICS

Unit-I: Introduction to Statistics

- Definition of statistics
- Types of data (quantitative, qualitative)
- Descriptive statistics (mean, median, mode, range, variance, standard deviation)
- Measures of central tendency and dispersion

Unit-II: Probability Theory

- Basic concepts of probability
- Probability rules (addition, multiplication, conditional probability)
- Discrete and continuous probability distributions (binomial, Poisson, normal)

Unit-III: Sampling and Estimation

- Sampling methods (simple random sampling, stratified sampling, cluster sampling)
- Point estimation (mean, proportion)

- Interval estimation (confidence intervals)

Unit-IV: Hypothesis Testing

- Hypothesis formulation
- Test statistics
- P-value and critical value approaches
- Hypothesis testing for means and proportions

Unit-V: Correlation and Regression

- Correlation (Pearson's correlation coefficient)
- Simple linear regression
- Multiple linear regression
- Model evaluation and interpretation

Unit-VI: Nonparametric Statistics

- Nonparametric tests (sign test, Wilcoxon rank-sum test, Kruskal-Wallis test)
- Applications of nonparametric methods

Unit-VII: Time Series Analysis

- Time series components (trend, seasonality, cycle, irregular)
- Forecasting methods (moving average, exponential smoothing, ARIMA)

Unit-VIII: Statistical Software Applications

- Introduction to statistical software (e.g., R, Python, SPSS)
- Data analysis using software

5. Educational Psychology

UNIT –I

Psychology of teaching: meaning, scope, Indian & Western concept and its relevance from the point of view of teachers, Stages in teaching: Pre active, Interactive & Post active, Levels in teaching: memory level, understanding level and reflective level. Psychology of development: meaning of development, difference between development & growth, characteristics related to cognitive, social and emotional development during childhood &adolescence and their educational implications.

UNIT –II

- Psychology of learning: meaning, factors affecting learning, transfer of learning: meaning, types and its educational implications, Motivation: meaning, various techniques for motivating the students.
- Theories of learning: Classical, Operant, Gestalt and Cognitive theory of Piaget and Bruner, Main features and their educational implications.

UNIT –III

- Psychology of intelligence: meaning, concept, theories of intelligence: cognitive, associative and punitive. IQ, EQ and SQ: their implications for organizing teaching learning processes. Use of intelligence tests and its limitations. Indian concept of intelligence.
- Psychology of Adjustment: meaning, process of adjustment, characteristics of a well-adjusted person. Stress: concept, coping mechanism and its educational implications for teacher & learner. Ensuring wellness life style: determinants of wellness and scales to measure wellness life style.

UNIT –IV

- Psychology of personality: meaning, its Indian and Western conceptualizations: development of personality, measurement of personality.
- Psychology of exceptional children: creative, gifted, slow learner and mentally retarded children their characteristics and implications for providing education.

6. Environmental Education

UNIT –I Nature and scope of environmental education

- Nature, need and scope of environmental education and its conservation
- Environmental education: a way of implementing the goals of environmental protection.
- Present status of environmental education at various levels India as a mega biodiversity Nation, Different ecosystems at national and global level.
- Role of individual in conservation of natural resources: water, energy and food
- Role of individual in prevention of pollution: air and water
- Equitable uses of resources for sustainable livelihoods
- Environmental legislation: awareness and issues involved in enforcement Role of information technology and media in environment awareness/consciousness

UNIT –II Community Participation and Environment

- Community participation in natural resource management ñ water, forests, etc.
- Change in forest cover over time
- Deforestation in the context of tribal life
- Sustainable land use management Traditional knowledge and biodiversity conservation
- Developmental projects, including Government initiatives and their impact on biodiversity Conservation
- Issues involved in enforcement of environment legislations
- Role of media and ecotourism in creating environmental awareness
- Role of local bodies in environmental management
- Shifting cultivation and its impact on environment

UNIT –III Environmental Issues and concerns

- Consumerism and waste generation and its management
- Genetically-modified crops and food security: Impacts positive and negative Water consumption pattern in rural and urban settlement
- Ethno-botany and its role in the present-day world
- Environmental degradation and its impact on the health of people
- Economic growth and sustainable consumption
- Organic farming
- Agricultural waste: Their impact and management
- Rain water harvesting and water resource management
- Biomedical waste management
  Changing patterns of energy and water consumption.

UNIT IV Initiatives by various agencies for Environment Education

- Environmental conservation in the globalised world in the context of global problem
- Alternative sources of energy
- Impact of natural-disaster/man-made disaster on environment
- Biological control for sustainable agriculture
- Heat production and greenhouse gas emission
- Impact of industry/mining/transport on environment
- Sustainable use of forest produces.
- Governmental and non-government initiatives.
- Supreme Court order implementation of Environmental Education (EE)

7.PRACTICAL-III

Unit I – Basic Concepts of C++

- Introduction to C++ environment, I/O using cin and cout.
- **Experiments / Sample Programs:**

1. Program to display a simple output message (“Hello C++”).
2. Program to demonstrate variables, constants, and scope.
3. Program to illustrate operators (arithmetic, relational, logical).
4. Program to swap two numbers using a function.
5. Program to demonstrate inline functions.

Unit II – Control Structures and Functions

- Decision making, looping, and function overloading.
- **Experiments / Sample Programs:**
    1. Program to check whether a number is even or odd using if-else.
    2. Program to find the largest of three numbers using nested if.
    3. Program to implement a simple calculator using switch-case.
    4. Program to generate factorial and Fibonacci series using functions.
    5. Program to demonstrate function overloading.
    6. Program to illustrate default arguments in functions.

Unit III – Arrays, Strings, and Pointers

- Arrays, character strings, and basic pointer operations.
- **Experiments / Sample Programs:**
    1. Program to find the sum and average of array elements.
    2. Program to sort elements of an array.
    3. Program to perform matrix addition and multiplication.
    4. Program to reverse a string and check palindrome.
    5. Program to illustrate pointer operations (pointer arithmetic, pointer to array).
    6. Program to swap values using call by reference.

Unit IV – Classes and Objects

- Defining classes, creating objects, constructors, destructors, member functions.
- **Experiments / Sample Programs:**
    1. Program to define a class Student with data members and functions to read and display details.

2. Program to define a class Complex with member functions to add and subtract complex numbers.
3. Program to demonstrate constructors and destructors.
4. Program to illustrate passing objects as function arguments.
5. Program to demonstrate static data members and static functions.

Unit V – Inheritance

- Single, multiple, multilevel, hierarchical, and hybrid inheritance.
- **Experiments / Sample Programs:**
    1. Program to demonstrate single inheritance (base and derived class).
    2. Program to demonstrate multiple inheritance.
    3. Program to illustrate multilevel inheritance.
    4. Program to show hierarchical inheritance with different derived classes.
    5. Program to demonstrate hybrid inheritance.
    6. Program to illustrate constructor invocation in inheritance.

Unit VI – Polymorphism

- Function overriding, virtual functions, and abstract classes.
- **Experiments / Sample Programs:**
    1. Program to demonstrate function overriding.
    2. Program to illustrate virtual functions and runtime polymorphism.
    3. Program to create an abstract base class and derive subclasses.
    4. Program to demonstrate pointer to base class accessing derived class objects.
    5. Program to illustrate pure virtual functions.

Unit VII – Operator Overloading and Friend Functions

- Overloading operators, using friend functions.
- **Experiments / Sample Programs:**
    1. Program to overload unary operators (++/--).
    2. Program to overload binary operators (+, -, *).

3. Program to overload relational operator (== or >).
4. Program to demonstrate friend function for two classes.
5. Program to demonstrate operator overloading using friend functions.

Unit VIII – Templates and Exception Handling

- Function templates, class templates, exceptions.
- **Experiments / Sample Programs:**
    1. Program to implement a function template for swapping two values.
    2. Program to implement a class template for a stack.
    3. Program to implement a class template for a queue.
    4. Program to demonstrate simple exception handling (try, catch, throw).
    5. Program to demonstrate multiple catch statements.

TAMILNADU COUNCIL FOR OPEN AND DISTANCE LEARNING
Approved by International Council for Open & Distance Education (ICDE), Oslo, Norway
Internationally Accredited Institution Registered under Tamilnadu Govt Act

INTEGRATED B.Sc.B.Ed. (Computer Science)

FOURTH YEAR

S.No	SUBJECT CODE	SUBJECT	MAX.MARKS
1	IBED-CS15	Operating Systems	100
2	IBED-CS16	Database Management Systems	100
3	IBED-CS17	Programming in Python	100
4	IBED-CS18	Assessment of Learning	100
5	IBED-CS19	Teaching Computer Science	100
6	IBED-CS20	Micro Teaching	100
7	IBED-CS21	Practical-IV	100

1.OPERATING SYSTEMS

Unit-I: Introduction to Operating Systems

- Definition and functions of operating systems
- Types of operating systems (batch, interactive, real-time)
- System calls and system programs

Unit-II: Process Management

- Process concepts (process states, process control block)
- Process scheduling (FCFS, SJF, priority scheduling, round robin)
- Process synchronization (critical section problem, mutexes, semaphores, monitors)
- Deadlock (detection, prevention, avoidance)

Unit-III: Memory Management

- Memory hierarchy (cache, main memory, secondary storage)
- Memory allocation (contiguous allocation, segmentation, paging)
- Virtual memory (paging, segmentation)
- Page replacement algorithms (FIFO, LRU, optimal)

Unit-IV: File Systems

- File concepts (file structure, file operations)

- File system implementation (directory structures, allocation methods)
- Disk scheduling algorithms (FCFS, SSTF, SCAN, C-SCAN)

Unit-V: I/O Systems

- I/O devices and their characteristics
- I/O techniques (programmed I/O, interrupt-driven I/O, DMA)
- Device drivers and device management

Unit-VI: Distributed Systems

- Introduction to distributed systems
- Distributed file systems
- Distributed process management
- Distributed synchronization

Unit-VII: Case Studies

- Analysis of popular operating systems (Linux, Windows, UNIX)
- Operating system design principles and trends

2. DATABASE MANAGEMENT SYSTEMS

Unit-I: Introduction to Database Systems

- Database Concepts: Definition, purpose, components, characteristics of a database.
- Database Management System: Definition, functions, advantages, disadvantages.
- Data Models: Hierarchical, network, relational, object-oriented, NoSQL.
- Data Independence: Physical and logical data independence.
- Database Life Cycle: Requirements analysis, conceptual design, logical design, physical design, implementation, operation, maintenance.

Unit-II: Relational Database Design

- Relational Model: Basic concepts (tuples, attributes, domains, keys), relational algebra.
- Normalization: First, second, and third normal forms, Boyce-Codd normal form.
- Denormalization: Reasons for denormalization, performance implications.

- Entity-Relationship (ER) Model: Entities, attributes, relationships, cardinality, and participation.
- ER to Relational Mapping: Converting ER diagrams into relational schemas.

Unit-III: SQL (Structured Query Language)

- Data Definition Language (DDL): Creating, altering, and dropping tables, indexes, and views.
- Data Manipulation Language (DML): Inserting, updating, deleting, and selecting data.
- Aggregate functions: COUNT, SUM, AVG, MIN, MAX.
- Joins: Inner, outer (left, right, full), self joins.
- Subqueries: Correlated and non-correlated subqueries.
- Views: Creating and using views.
- Constraints: Primary key, foreign key, unique, check, not null.

Unit-IV: Transaction Management

- Transactions: Definition, properties (ACID), states of a transaction.
- Concurrency Control: Serializability, conflicts, locking (two-phase locking, deadlock prevention).
- Recovery Management: Log-based recovery, checkpoints, recovery techniques.

Unit-V: Database Design and Implementation

- Database Design Methodology: Waterfall, prototyping, agile.
- Physical Design: Indexing techniques, query optimization.
- Database Implementation: Database software selection, installation, and configuration.
- Performance Tuning: Techniques for improving database performance.

Unit-VI: Advanced Topics

- NoSQL Databases: Key-value, document, graph, wide-column stores.
- Data Warehousing and Data Mining: Concepts, OLAP, data mining techniques.
- Distributed Databases: Replication, partitioning, fragmentation.
- Object-Relational Databases: Combining relational and object-oriented models.

3.PROGRAMMING IN PYTHON

Unit-I: Introduction to Python

- Python Overview: History, philosophy, features, applications.
- Installation and Setup: Installing Python, using IDLE or other IDEs.
- Basic Python Syntax: Variables, data types (numbers, strings, lists, tuples, dictionaries, sets), operators.
- Control Flow: Conditional statements (if, else, elif), loops (for, while).
- Functions: Defining, calling, parameters, return values.

Unit-II: Data Structures and Algorithms

- Lists: Operations, slicing, list comprehensions.
- Tuples: Immutability, unpacking.
- Dictionaries: Key-value pairs, methods.
- Sets: Operations, membership testing.
- Algorithms: Sorting (bubble, insertion, selection, merge, quick), searching (linear, binary).

Unit-III: Object-Oriented Programming (OOP)

- Classes and Objects: Creating classes, instances, attributes, methods.
- Inheritance: Single and multiple inheritance, polymorphism.
- Encapsulation: Public, private, and protected access modifiers.
- Abstraction: Abstract classes and interfaces.

Unit-IV: File I/O and Exception Handling

- File Operations: Reading, writing, appending files.
- Exception Handling: Try-except blocks, raising exceptions.
- File Formats: CSV, JSON, XML.

Unit-V: Modules and Packages

- Modules: Importing modules, creating custom modules.
- Packages: Organizing modules into packages.
- Standard Library: Exploring built-in modules (math, random, time, etc.).

Unit-VI: Advanced Topics

- Regular Expressions: Pattern matching and text manipulation.
- Functional Programming: Lambda functions, map, filter, reduce.
- Decorators: Modifying functions and methods.
- Generators: Creating iterators using yield.
- Networking: Using sockets for client-server communication.

Unit-VII: GUI Programming (Optional)

- Tkinter: Introduction to Tkinter, creating GUI applications.
- Other GUI Frameworks: PyQt, wxPython (brief overview).

4. Assessment of Learning

Unit –I

- Overview of Assessment and Evaluation
- Perspective on assessment and evaluation of learning in a constructivist paradigm
- Distinction between 'Assessment of Learning' and 'Assessment for Learning'
- Purpose of assessment in a 'constructivist' paradigm
- Engage learners' minds in order to further learning in various dimensions.
- Promote development in cognitive, social and emotional aspects.
- Developing distinctions between the terms
- assessment, evaluation, test, examination, measurement
- formative and summative evaluation
- continuous and comprehensive assessment
- Understanding notions of 'Subject-based Learning' in a constructivist Perspective

Unit –II

- Dimensions to consider for Assessment
- Dimensions and levels of learning
- Retention/recall of facts and concepts; Application of specific skills
- Manipulating tools and symbols; Problem-solving; applying learning to diverse situations
- Meaning-making propensity; Abstraction of ideas from experiences;
- Seeing links and relationships; Inference; Analysis; Reflection
- Originality and initiative, Collaborative participation, Creativity, Flexibility

- Contexts of assessment- Subject-related, Person-related

Unit –III

- Teacher Competencies in Evolving Appropriate Assessment Tools
- Visualizing appropriate assessment tools for specific contexts, Content, and student
- Formulating tasks and questions that engage the learner and demonstrate the process of thinking; Scope
- for original responses
- Evolving suitable criteria for assessment
- Organizing and planning for student portfolios and developing rubrics for portfolio assessment
- Using assessment feedback for furthering learning.

Unit-IV

- Examination System: Reforms
- Place of marks, grades and qualitative descriptions
- Examination for social selection and placement
- Introducing flexibility in examination-taking requirements
- Improving quality and range of questions in exam papers school-based credits
- Examination management
- Role of ICT in examination

Unit-V

- Data Analysis, Feedback and Reporting
- Statistical tools, Percentage, graphical representation, frequency distribution, central
- tendency, variation, normal distribution, percentile rank, correlation and their interpretation
- Graphical representation of results.
- Feedback as an essential component of formative assessment
- Use of assessment for feedback; For taking pedagogic decisions
- Types of teacher feedback (written comments, oral); Peer feedback
- Developing and maintaining a comprehensive learner profile
- Purposes of reporting: To communicate
- Progress and profile of learner
- Basis for further pedagogic decisions
- Reporting a consolidated learner profile.

5.Teaching Computer Science

Unit-I

Nature And Scope of Computer Science Meaning and introduction of computer-Importance of Computer Science-Computer curriculum-Its place relationship with other subjects-Interdisciplinary approaches in school curriculum-Various branches of computer science-Computer generations-types of computers-Significant discoveries-Hardware-Software- Programming Languages-Computer Network-Computer virus and protection-Use of computers in education.

Unit–II

Aims and Objectives of Teaching Computer Science Aims and objectives of teaching computer science-Bloom's Taxonomy: Cognitive, Affective and Psychomotor Domains-Computer science teaching at different levels: Primary, Secondary and Higher Secondary Attainment of the objectives of computer science teaching.

Unit–III

Microteaching Microteaching-Definition, origin, need, & procedure-Microteaching Cycle-Communication skills: verbal and non-verbal communication-Types of Skills-Skill of Introducing, Explaining, Stimulus variation, Questioning, Demonstration, reinforcement, verbal and non-verbal cues, Illustration with examples, and Closure Link Practice - Need for Link practice.

Unit-IV

Planning For Teaching- Learning of Computer Science Lesson Planning-Importance of lesson plans-Writing Instructional Objectives-Planning for specific behavioural changes-Preparation and use of Unit Plan-Teaching and Teaching Aids-Evaluation, Recapitulation and Assignments.

Unit-V

Evaluation in Computer Science Difference between Measurement, Assessment and Evaluation-Characteristics of good Measurement-Diagnostic Test and Remedial Teaching-Criterion Referenced Testing and Norm Referenced Testing-Different types of items-Multiple Discriminant Type Item-Development and Standardization of Achievement Test in Computer Science.

6. Microteaching

Step 1: Select the Topic

- Choose a topic from your specialization/subject area that is suitable for 10–15 minutes of teaching.
- Ensure the topic is clear, focused, and limited in scope.
- Check that necessary teaching materials/resources are available.

Tips:

- Pick a concept that can be explained clearly in a short time.
- Avoid overly broad topics that cannot be completed in the session.

Step 2: Analyse Learners

- Identify the target learners: age, grade, prior knowledge, and learning needs.
- Consider what students already know about the topic and what misconceptions may exist.

Questions to consider:

- What is the entry behaviour of learners?
- What learning objectives can realistically be achieved?

Step 3: Set Objectives

- Write specific, measurable, and achievable learning objectives.
- Use Bloom's Taxonomy verbs (e.g., identify, explain, demonstrate, analyse).

Example:

- *By the end of the lesson, students will be able to identify three types of clouds and describe their characteristics.*

Step 4: Plan the Lesson

1. Introduction (2–3 minutes):
    - Capture attention (question, story, demonstration).
    - State the objectives of the lesson.
2. Content Development (3–5 minutes):
    - Present content in a logical sequence.
    - Use examples, diagrams, and real-life applications.
    - Check understanding by asking short questions.

3. Conclusion (1–2 minutes):
    - Summarize key points.
    - Reinforce learning with a quick recap or mini-activity.
    - Provide closure to the lesson.

Step 5: Prepare Teaching Materials

- Create teaching aids according to the topic and student needs.
- Examples:
    - Charts, flashcards, slides (PowerPoint/Google Slides)
    - Models, props, real objects
    - Whiteboard illustrations or handouts

Tip: Keep materials simple and relevant; avoid overloading.

Step 6: Practice the Lesson

- Rehearse the lesson several times.
- Time yourself to ensure it fits the micro-teaching duration.
- Practice body language, voice modulation, and gestures.

Optional: Record yourself to evaluate clarity, pace, and engagement**.**

Step 7: Deliver the Lesson

- Start confidently with a greeting and introduction.
- Follow your lesson plan.
- Maintain eye contact, gestures, and engagement.
- Ask questions to check learner understanding.
- Conclude by summarizing key points and objectives.

Step 8: Use Feedback

- After delivery, receive feedback from peers and instructors.
- Focus on:
    - Content clarity
    - Use of teaching aids
    - Interaction with learners
    - Pace, voice, and body language
- Reflect on strengths and areas for improvement.

Step 9: Revise and Reflect

- Note down feedback and plan improvements.
- Modify teaching style, materials, or explanation methods for future lessons.
- Keep a micro-teaching record with objectives, plan, delivery, and reflections.

7.Practical-IV

Unit I – Basics of Python Programming

- Introduction to Python environment, simple I/O, variables, and operators.
- **Experiments / Sample Programs:**
    1. Program to display a welcome message.
    2. Program to demonstrate variables and basic data types.
    3. Program to perform arithmetic operations.
    4. Program to calculate simple interest and area of geometric shapes.
    5. Program to convert temperature (Celsius ↔ Fahrenheit).

Unit II – Control Structures

- Conditional statements and looping constructs.
- **Experiments / Sample Programs:**
    1. Program to check whether a number is even or odd.
    2. Program to find the largest of three numbers.
    3. Program to check leap year.
    4. Program to display multiplication table using for loop.
    5. Program to generate factorial and Fibonacci series.
    6. Program to check for prime numbers.
    7. Pattern generation using nested loops.

Unit III – Data Structures: Lists, Tuples, Sets, Dictionaries

- Usage of built-in collections for data handling.
- **Experiments / Sample Programs:**
    1. Program to find the sum and average of elements in a list.

2. Program to sort a list and perform searching.
3. Program to demonstrate list slicing and list comprehensions.
4. Program to demonstrate tuple operations.
5. Program to perform set operations (union, intersection, difference).
6. Program to create and manipulate dictionaries (insert, update, delete, search).

Unit IV – Strings

- String operations and manipulations.
- **Experiments / Sample Programs:**
    1. Program to find the length of a string without using built-in functions.
    2. Program to reverse a string.
    3. Program to check if a string is a palindrome.
    4. Program to count vowels, consonants, digits, and spaces in a string.
    5. Program to demonstrate string functions (split, join, replace, find).

Unit V – Functions and Recursion

- User-defined functions, default arguments, recursion.
- **Experiments / Sample Programs:**
    1. Program to find factorial using recursion.
    2. Program to generate Fibonacci sequence using recursion.
    3. Program to demonstrate default and keyword arguments.
    4. Program to illustrate variable scope (local, global, nonlocal).
    5. Program to pass lists/tuples to functions.

Unit VI – Modules and Packages

- Importing built-in and user-defined modules.
- **Experiments / Sample Programs:**
    1. Program to demonstrate use of math module functions.
    2. Program to generate random numbers using random module.
    3. Program to display calendar for a given month and year.

4. Program to create a user-defined module and import it.
5. Program to demonstrate use of packages.

Unit VII – Object Oriented Programming in Python

- Classes, objects, inheritance, polymorphism.
- **Experiments / Sample Programs:**
    1. Program to define a class Student and display student details.
    2. Program to implement a class with constructor and destructor.
    3. Program to illustrate single and multiple inheritance.
    4. Program to demonstrate operator overloading using special methods.
    5. Program to illustrate polymorphism using method overriding.

Unit VIII – Exception Handling and File Handling

- Error handling and persistent data storage.
- **Experiments / Sample Programs:**
    1. Program to demonstrate try-except blocks.
    2. Program to demonstrate multiple exception handling.
    3. Program to read from a file and display contents.
    4. Program to write and append data to a file.
    5. Program to count words, characters, and lines in a file.
    6. Program to copy contents from one file to another.