**LESSON PLAN 2021-22(ODD SEM) BCA-352: Operating System-I**

**OCTOBER-21**

Operating System: Definition, Characteristics, Components, Functions, Examples; Types of Operating System: Single User/Multi User, Classification of Operating System: Batch, Multiprogrammed, Timesharing, Multiprocessing, Parallel, Distributed, Real Time;

**NOVEMBER-21**

System Calls and System Programs: Process Control, File Manipulation, Device Manipulation, Information Maintenance, Communications, Process Management: Process concept, Process states and Process Control Block; Process Scheduling: Scheduling Queues, Schedulers, Context Switch;

**DECEMBER-21**

Operation on Processes: Process Creation, Process Termination; Cooperating Processes, Introduction to Threads, Inter-process Communication; CPU Scheduling: Basic Concepts, Scheduling Criteria, Scheduling Algorithms: FCFS, SJF, Priority, Round-Robin, Multilevel Queue, Multilevel Feedback Queue Scheduling

**JANUARY-22**

Deadlocks: System Model, Deadlock Characterization, Methods of Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection and Recovery Memory Management: Introduction, Swapping, Contiguous Allocation: Single-Partition/Multiple Partition Allocation, External/Internal Fragmentation; Paging: Basic Method, Hardware, Implementation of Page table; Segmentation: Basic Method, Hardware, Implementation of Segment Table, Advantages/Disadvantages of Paging/Segmentation

**FEBRUARY-22**

Virtual Memory: Introduction, Demand Paging, Page Replacement, Page Replacement Algorithms: FIFO, Optimal, LRU, Counting; Thrashing and its cause; File Management: File Concepts, File Attributes, File Operations, File Types, File Access/Allocation Methods, File Protection, File Recovery.

**LESSON PLAN 2021-22(ODD SEM) BCA-355: Programming Using Visual Basic**

**OCTOBER**

Introduction to VB: Visual & Non-Visual programming, Procedural, Object-Oriented, Object-Based and Event-Driven Programming Languages, VB as Even-Driven and Object-Based Language, VB Environment: Menu bar, Toolbar, Project explorer, Toolbox, Properties Window, Form Designer, Form Layout, Immediate window, Default Controls in Tool Box Visual Development and Event Driven programming

**NOVEMBER**

Basics of Programming: Variables: Declaring Variables, Types of variables, Converting Variables Types, User Defined Data Types, Forcing Variable Declaration, Scope & Lifetime of Variables, Constants: Named & Intrinsic, Operators: Arithmetic, Relational & Logical operators, Input/output in VB: Various Controls for I/O, Message box, Input Box, Print statement,

**DECEMBER**

Decision Statements in VB - if statement, if-then-else, select-case; Looping Statements in VB: do-loop, for-next, while-wend; Exit statement, Nested Control Structure; Arrays: Declaring and using Arrays, One-dimensional, Two-dimensional and Multi-dimensional Arrays, Static and Dynamic arrays, Array of Arrays. Simple Program Development in VB such as Sum of Numbers, Greatest among Numbers, Checking Even/Odd Number, HCF of Two Numbers,

**JANUARY**

Procedures: General & Event Procedures, Subroutines, Functions, Calling Procedures, Arguments - Passing Mechanisms, Optional Arguments, Named Arguments, Functions Returning Custom Data Types

**FEBRUARY**

Generate Prime Numbers; Generate Fibonacci Series, Factorial of a Number, Searching, Sorting, etc.

**LESSON PLAN 2021-22(ODD SEM) BCA – 231 OBJECT ORIENTED PROGRAMMING USING ‘C++**

**OCTOBER**

Object oriented Programming: Object-Oriented programming features and benefits. Object-Oriented features of C++, Class and Objects, Data Hiding & Encapsulation, Structures, Data members and Member functions, Scope resolution operator and its significance, Static Data Members, Static member functions, Nested and Local Class, Accessing Members of Class and Structure

**NOVEMBER**

Constructor, Initialization using constructor, types of constructor– Default, Parameterized & Copy Constructors, Constructor overloading, Default Values to Parameters, Destructors, Console I/O: Hierarchy of Console Stream Classes, Unformatted and Formatted I/O Operations

**DECEMBER**

Manipulators, Friend Function, Friend Class, Arrays, Array of Objects, Passing and Returning Objects to Functions, String Handling in C++,

**JANUARY**

Dynamic Memory Management: Pointers, new and delete Operator, Array of Pointers to Objects, this Pointer, Passing Parameters to Functions by Reference & pointers, Polymorphism: Operators in C++, Precedence and Associativity Rules, Operator Overloading, Unary & Binary Operators Overloading,

**FEBRUARY**

Function Overloading, Inline Functions

**LESSON PLAN 2021-22(ODD SEM) BCA – 232 DATA STRUCTURES**

**OCTOBER**

Introduction: Elementary data organization, Data Structure definition, Data type vs. data structure, Categories of data structures, Data structure operations, Applications of data structures, Algorithms complexity and time-space tradeoff, Big-O notation.

**NOVEMBER**

Strings: Introduction, String strings, String operations, Pattern matching algorithms. Arrays: Introduction, Linear arrays, Representation of linear array in memory, Traversal, Insertions, Deletion in an array, Multidimensional arrays, Parallel arrays, Sparce matrics.

**DECEMBER**

Linked List: Introduction, Array vs. linked list, Representation of linked lists in memory, Traversal, Insertion, Deletion, Searching in a linked list, Header linked list, Circular linked list, Two-way linked list, Garbage collection, Applications of linked lists. Algorithms for Insertion, deletion in array, Single linked list

**JANUARY**

Stack: Introduction, Array and linked representation of stacks, Operations on stacks, Applications of stacks: Polish notation, Recursion. Queues: Introduction, Array and linked representation of queues, Operations on queues, Deques, Priority Queues, Applications of queues.

**FEBRUARY**

Tree: Introduction, Definition, Representing Binary tree in memory, Traversing binary trees, Traversal algorithms using stacks and using recursion. Graph: Introduction, Graph theory terminology, Sequential and linked representation of graphs.