

**B.Sc. II YEAR Computer Science (For Regular Students)**

Paper	Subjects	Internals			Theory	Total (B)	Total A+B	Practical	Grand Total
		3 Months	6 Months	Total (A)					
I	Object Oriented Programming concepts using C++ and Java	10	10	20	40	80	100	50	150
II	Data Structure and Software Analysis and Engineering				40				

## BSC II YEAR

### Paper-I Object Oriented Programming concepts using C++ and Java

**MAX:40**

**MIN:13**

#### UNIT I

Introduction to C++: programming paradigms, Key concepts of object-oriented programming, advantages of OOP's. Input and output in C++: pre-defined streams, unformatted console I/O operations, formatted console I/O operations. C++ declarations: parts of C++ program, types of tokens, keywords, identifiers, data types, constants, operators, precedence of operators, referencing and dereferencing operators, scope access operator. Control structures: decision making statements, looping statements.

#### UNIT II

Functions: main(), parts of function, passing arguments: value, address, reference, inline functions, function overloading: principles, precautions, library functions. Classes and objects: declaring classes and objects, accessing class members, keyword: public, private, protected, defining member function: member function inside the class, member function outside the class, static member functions variables and function, friend function, friend class, overloading member function. Constructors and Destructors: characteristics, applications, constructors with arguments, overloading constructors, types of constructors.

#### UNIT III

Operator overloading: overloading unary operator, binary operator. Inheritance: access specifiers: public inheritance, private inheritance, protected data with private inheritance, types of inheritances: single, multiple, hierarchical, multilevel, hybrid, multipath, virtual base class. Pointers & arrays: pointer declaration, pointer to class & object, Array: declarations & initialization, arrays of classes. Polymorphism: static (Early) binding, dynamic (Late) binding, virtual function, pure virtual function.

#### UNIT IV

C++ vs JAVA, JAVA environment, Structure of Java, Tokens, Statements, JAVA virtual machine. Define a Class, Adding Variables and Methods, Creating Objects, Accessing Class Members, Constructors, Method Overloading, Static Members, Nesting of Methods. Arrays: One Dimensional & two Dimensional, strings, Vectors, wrapper Classes. Define Interface, Extend Interface, Implement Interface, Accessing Interface Variables. System Packages, Using System Package, adding a Class to a Package, Hiding Classes.

#### UNIT V

Inheritance: Extending a Class, Overriding Methods, Final Variables and Methods, Final Classes, Finalize Methods, Abstract methods and Classes, Visibility Control. Creating Threads, Extending the Threads Class, Stopping and Blocking a Thread, Life Cycle of a Thread, Using Thread Methods, Thread Exceptions, Thread Priority, Synchronization, Implementing the Runnable Interface. Writing Applets, Applets Life Cycle, Creating an

Executable Applet, designing a Web Page, Applet Tag, Adding Applet to HTML File, Running the Applet, Passing Parameters to Applets, Aligning the Display, HTML Tags & Applets, Getting Input from the User.

**Text Books:**

Object-Oriented programming with ANSI & turbo C++ by Ashok N. Kamthane.

Object-Oriented programming in C++ by E.Balagurusamy

E. Balaguruswamy, "Programming In Java", 2nd Edition, TMH Publications ISBN 0-07-463542-5

**Reference Books:**

C+ Object-Oriented programming in C++ by Robert Lafore.

The complete reference by Herbert Schildt, TMH publication.

**BSC II Year**  
**Paper-II Data Structure and Software Analysis and Engineering**

**MAX:40**

**MIN:13**

**UNIT I**

Concept of data structure and analysis of algorithm, abstract data structure, Introduction to stack and primitive operation on stack using array. Stack applications:-Infix Prefix, Postfix and Recursion, Introduction to queues, Primitive operation on queues using array, circular queue and applications of queue.

**UNIT II**

**LINKED LIST-** Introduction to Linked List, Types of Linked List (Singly, Circular Linked List, Doubly Linked List), Basic operations on Linked List (Insertion, Deletion & Traverse), Stacks, Queues, Circular Queue using Singly Linked List and Application of Linked List.

**UNIT III**

**TREES-**Basic terminology ,Binary Trees, Tree representations as array and Linked List, Basic operation Binary tree, Traversal of Binary trees:- In-order, Preorder, Post order, Application of Binary tree, Threaded Binary tree.

**SEARCHING & SORTING:** Sequential Searching, Binary search, Insertion sort, Selection sort, Quick sort, Bubble sort, Heap sort, Comparison of sorting methods.

**GRAPH:** Introduction of graphs, definition, Terminology, Directed, Undirected and weighted graph, representation of graph, graph traversal-Depth first, breath first search, Spanning tree, Minimum spanning tree (kruskal , prim's), shortest path algorithm.

**Unit IV**

**INFORMATION TO SOFTWARE ENGINEERING:** Software Engineering- A Layered Technology, The Software Process, Software Process Models, The Linear Sequential Model, The Prototyping Model, The RAD Model, The Incremental Model, The Spiral Model, Comparison of different life-cycle models. Requirement Analysis and Specification: Properties of a good SRS document.

**DESIGN CONCEPTS AND PRINCIPLES:** Cohesion, Coupling and Functional Dependence, need of function independence, function-oriented design and object oriented design, DFD model of a system.

**Unit V:**

**PROJECT ESTIMATION:** Project planning, Metrics for software project size estimation: Lines of Code (LOC), Function point (FP), Limitations of function point (FP) metric.

**PROJECT ESTIMATION TECHNIQUES:** Empirical estimation techniques - COCOMO Model

**SOFTWARE TESTING FUNDAMENTALS:** Black Box Testing - equivalence class partitioning, Boundary Value Analysis. White Box Testing, Control flow graph, Cyclomatic complexity.

**SOFTWARE TESTING STRATEGIES:** Unit Testing, Regression Testing, System Testing: Alpha and Beta Testing, Integration testing, Recovery Testing, Stress Testing, Performance Testing.

### **Text Books**

1. Software Engineering: A Practitioner's Approach, Author: Roger S. Pressman
2. Data Structures Through C ( A practical Approach), Author: G.S. Baluja

### **Reference Books**

1. Data Structures using C++ Author: D.S. Malik Second Edition
2. Principles of Data Structures using C and C++ Author: Vinu V Das New Age International Publishers
3. A Practical Introduction to Data Structures and Algorithm Analysis Author: Clifford A. Snaffer Third Edition (Java)
4. Data Structures and Algorithms Made Easy Narasimha Karumanchi
5. An Integrated Approach to Software Engineering Author: Pankaj Jalote