# **BCSC 0020.: Object-Oriented Concepts Using Java**

# **Objective: The course enables students to understand virtualization technology, Applications along with cloud computing concepts and services and to study different cloud architecture & deployment models.**

|  |  |  |
| --- | --- | --- |
| **Module No.** | **Content** | **Teaching Hours** |
| **I** | **Algorithm:** Introduction, Features, Different Ways of Starting Algorithms.  **Flow Chart:** Introduction, Standard, Guidelines, Advantages, and Limitations of using Flowcharts.  **Basics of Java:** Overview, Structure of a Java program, Identifier, Keywords, Variables, Data types, Formatted Input, and Output.  **Operators and Expression:** Assignment, Arithmetic, Relational, Logical, Bitwise, Conditional. **Decision and Case Control Structure:** if, if-else, nested if-else **Loop Control Structure:** For loop, while loop, do-while loop, nesting of loops, break, and continue. **Data Types**: Primitive data types and non-primitive data types  IEEE representation of data types like float & double  **Type Conversion:** Type Promotion in expression, Conversion by Assignment, Truncation, and Casting Arithmetic expression. **Functions**: In this section, you will learn to write code in a modular way to help you reuse the code in different Java files and make it a more readable and organized, global variable  **Arrays:** Introduction, one-dimensional, Array-Declaration, Initialization, Address Calculation.  **Operations on Arrays:** Insertion, Deletion, Basic Implementation bases Question | 25 |
| **II** | **Object-Oriented Programming:** Features of Object-Oriented Programming, Introduction to Object-Oriented Java Programming.  **Java Technology & Environment:** Understanding the compilation process of the JVM, JVM vs JDK vs JRE, Key Features of Java, and Structure of a Simple Java Program.  **Working with Java Primitive Data Types:** Strongly Typed nature of Java, Primitive Data Types in Java, The new ‘var’ keyword, Scope of a variable.  **Accepting User Input in Java Programs:** using the Scanner class, using command line arguments.  **Programming Constructs:** Sequence, Selection, Iteration & Transfer Statements, For-Each Loop.  **Creating and Using Methods:** Signature of a method, Types of Methods, Overloading methods in a class, Static and Non-Static Methods.  **Describing and Using Objects & Classes:** Declare the structure of a Java class, declare members of a class (fields and methods), declare and use Java Objects, the lifecycle of an Object (creation, assignment, dereferencing, and garbage collection), Constructors of a class, Overloading Constructors, Constructor chaining using ‘this’ and ‘super’ keyword.  **Using Java Packages:** create and import Java packages and static imports, abstract program logic to packages, create executable main class, running the executable class inside a package.  **Applying Encapsulation**: Using access modifiers with/in a class, principles of encapsulation. | 25 |

**Reference Books/ Text Books / Cases:**

The Complete Reference Java by Herbert Shildt 3rd Edition, Tata McGraw Hill

**Intended Outcomes:**

1. Students should thoroughly understand object-oriented programming principles and techniques such as encapsulation, polymorphism, and inheritance and use them to develop rather complex programs using classes and objects in JAVA.

2. Students should be able to implement polymorphism features like function overloading.

3. Students will be able to program more advanced JAVA features such as the composition of objects, dynamic memory allocation, and inheritance.

4. Students should thoroughly understand the fundamental features of an object-oriented language Java and learn to implement interfaces, abstract classes, packages, etc.

5. Students should be able to implement Exception Handling and Multithreading in Java.