

|                                |   |                 |            |                |
|--------------------------------|---|-----------------|------------|----------------|
| <b>Course Code</b>             | <b>40110202</b>                                     |                 |            |                |
| <b>Category</b>                | <b>Core Subject</b>                                 |                 |            |                |
| <b>Course Title</b>            | <b>Object Oriented Programming with JAVA (OOPJ)</b> |                 |            |                |
| <b>Scheme and Credits</b>      | <b>Theory</b>                                       | <b>Tutorial</b> | <b>Lab</b> | <b>Credits</b> |
|                                | <b>3</b>  | <b>0</b>        | <b>4</b>   | <b>5</b>       |
| <b>Pre-requisites (if any)</b> | Knowledge of the C programming language             |                 |            |                |

### 1. Course Objectives:

|   |  |
|---|--|
| 1 | To understand concepts of Object Oriented Programming Language (OOPL)  |
| 2 | To differentiate between Object Oriented Programming Language and Procedure Oriented Programming Language (POPL) |
| 3 | To understand and implement class, inheritance & polymorphism concepts   |
| 4 | To implement error handling mechanism  |
| 5 | To implement multi-threaded applications   |
| 6 | To develop application using different file operations   |
| 7 | To develop GUI based application using swing components  |

### 2. Course Contents:

| <b>Unit</b>   | <b>Course Content</b>  | <b>Weightage</b> |
|---------------|--|------------------|
| <b>Unit I</b> | <p><b>Basics of OOPL &amp; Class:</b></p> <p><b>Basics:</b> Object-Oriented Programming concepts (object, class, encapsulation, abstraction, inheritance, polymorphism, message passing, dynamic binding), features of Java, Java's magic: bytecode</p> <p>(First day lab activity: Understanding Java Development Kit (JDK) and its settings, compiling and executing the first simple program, comments, Java class libraries, arrays, difference between C &amp; Java)</p> <p><b>Class:</b> Class fundamentals, the general form of a class, defining a class, creating objects, working with method, constructors (default, parameterized, copy constructor), garbage collector, this keyword, static block, initializer block</p> <p><b>A Closer Look at Methods and Classes:</b> Passing object to the method, returning object from the method, method overloading, constructor overloading, static variable, static method, nested and</p> | <b>25%</b>       |

|                 |  |            |
|-----------------|--|------------|
|                 | inner classes, local class, anonymous class  |            |
| <b>Unit II</b>  | <b>Inheritance &amp; Polymorphism:</b><br><b>Inheritance &amp; Polymorphism:</b> Inheritance, constructors and inheritance, types of inheritance, uses of super keyword for constructor and method, creating a multilevel hierarchy, constructor's execution sequence, superclass references and subclass objects, method overriding, polymorphism, abstract class, final keyword, Object class and its methods  | <b>20%</b> |
| <b>Unit III</b> | <b>Interfaces, Exception &amp; Collection:</b><br><b>Interfaces:</b> Interface, working with interface, multiple inheritance with interface<br><br><b>Exception:</b><br>Exception hierarchy, exception handling fundamentals, the consequences of an uncaught exception, using multiple catch statements, catching subclass exception, nested try blocks, re-throwing an exception, finally & throws keywords, java's built-in exceptions, user define exception<br><br><b>Collection Framework :</b> Collection, Collection framework, Collection interface, Iterator interface, list interface, stack interface, vector interface  | <b>15%</b> |
| <b>Unit IV</b>  | <b>Multi-Threading, Generic, I/O</b><br><b>Multithreaded Programming:</b><br>Multithreading fundamentals, the thread class and runnable interface, creating a thread, creating multiple threads, determining thread methods, thread priorities, need for synchronization, thread synchronization<br><br><b>Using I/O:</b> Working with File class, creating file & folder, Renaming file & folder, Deleting file & folder, byte streams and character streams, the byte stream classes, the character stream classes, the predefined streams, using the byte streams (reading console input, writing console output), reading and writing files using byte streams (inputting from a file, writing to a file), automatically closing a file, reading and writing binary data, random access file | <b>20%</b> |
| <b>Unit V</b>   | <b>GUI Programming with Swing:</b><br><b>Introduction:</b> GUI Programming, AWT Components, swing components, comparing AWT and swing, event handling mechanism<br><br><b>Swing components with event handling:</b> JFrame, JPanel, JButton, Layout Managers (FlowLayout, BorderLayout, GridLayout, GridBagLayout, CardLayout), JLabel, JTextField, JTextArea, JPasswordField, JCheckBox, JRadioButton, JComboBox, JList, JScrollBar, Adapter classes, dialog boxes, working with menus  | <b>20%</b> |

### 3. Text Books:

1. The Complete Reference – Java, 7<sup>th</sup> Edition, by Herbert Schildt, Tata McGraw Hill



Publication

2. Beginning Java 2 JDK, 5<sup>th</sup> Edition, by Ivor Horton - Wiley Publication
3. Core Java , Volume I – Fundamentals, 10<sup>th</sup> Edition, by Cay S. Horstmann- Pearson Publication
4. Java™: A Beginner’s Guide, 7<sup>th</sup> Edition, by Herbert Schildt - McGraw Hill Publication
5. Java Programming by Hari Mohan Pandey – 1<sup>st</sup> Edition, Pearson Publication
6. Programming With Java, 5<sup>th</sup> Edition, by E. Balagurusamy- McGraw Hill Publisher

**4. Webilography:**

1. <https://www.java.com/en/>
2. <https://docs.oracle.com/javase/8/docs/api/java/>
3. <https://www.w3schools.com/java>

**5. Tools: Latest Version of NetBeans IDE**

**6. Accomplishment of the student after completing the course:**

After completion of the course students should be capable of developing console based and GUI based desktop based application through Java programming language.