|  |  |
| --- | --- |
| **Course-19 Title: Objective Oriented Programming Sessional** |  |
| **Course No.:CCE-122 Credit : 1.5 Contact Hours: 3** | **Total Marks: 100** |

**11.1 Rationale:**

This course is designed to teach Object-Oriented programming concepts, techniques, and applications using the Java programming language.

**11.2 Objectives:**

1. The **model** of object oriented programming: **abstract data types, encapsulation, inheritance** and **polymorphism**
2. **Fundamental features** of an object oriented language like Java: **object classes** and **interfaces, exceptions** and libraries of **object collections**
3. How to take the statement of a business problem and from this determine **suitable logic for solving the problem;** then be able to proceed to code that logic as a program written in Java.
4. How to **test, document** and **prepare** a professional looking package for each business project using javadoc.

|  |  |  |  |
| --- | --- | --- | --- |
| **11.3**  **Learning Outcomes** | **11.4**  **Course Content** | **11.5**  **Teaching Strategy/ Learning Experience** | **11.6 Assessment Strategy** |
| * Basic programming | Study basic concept of OOP | **Exercise** | **Practice and coding** |
| * Write OOP program | Study access modifier, class creation, object creation | **Exercise** | **Practice and coding** |
| * Simple logic implementation in OOP | Study different control structure, event handling | **Exercise** | **Practice and coding** |
| * Create OOP program with OOP feature * Learn Array programming | Study consturctor,overloaded constructor,array programming | **Exercise** | **Practice and coding** |
| * Implement interface in writing of OOP program | Study interface | **Exercise** | **Practice and coding** |
| * Implement inheritance in writing of OOP program | Study inheritance | **Exercise** | **Practice and coding** |
| * Implement polymorphism in writing of OOP program | Studfy method overloading and method overload | **Exercise** | **Practice and coding** |
| * Handling exception | Study different types of built in and custom exception | **Exercise** | **Practice and coding** |
| * Apply string to solve big mathematical problem like big integer problem | Study string manipulation | **Exercise** | **Practice and coding** |
| * Create GUI based program | Study GUI | **Exercise** | **Practice and coding** |
| * Apply threading in writing advanced OOP program | Study threading and client server architecture | **Exercise** | **Practice and coding** |

**RECOMMENDED BOOKS AND PERIODICALS**

1. Java™ How to Program, Latest Edition **by** H. M. Deitel -  Deitel & Associates, Inc., P. J. Deitel -  Deitel & Associates, Inc

2. Liang, Y. D. (2005), Introduction to Java Programming, Comprehensive: International Edition, 5/E, Prentice Hall. ISBN 0-13-185721

Additional Useful Texts

Barnes, D. & Kolling, M. 2005, Objects First with Java: A Practical Introduction using BlueJ, 2nd edn, Pearson.

Farrell, J. 2005, Programming Logic and Design, Comprehensive, 3rd edn, Thomson.

Malik, D. & Nair, P. 2003, Java Programming: From Problem Analysis to Program Design, Thomson.

Robertson, L. 2003, Simple Program Design: A Step-by-Step Approach, 4th edn, Thomson.

Russell, J. 2001, Java Programming for the Absolute Beginner, Premier Press. Shelly, G., Cashman, T., Starks, J. & Mick, M. 2004, Java Programming Comprehensive Concepts and Techniques, Thomson.

Sun Java technology: http://java.sun.com/

Borland JBuilder: http://www.borland.com/jbuilder

Web page for the prescribed text:

http://www.cs.armstrong.edu/liang/intro5e/intro5estudentsolution.html