# Aror University of Art, Architecture, Design & Heritage Sukkur.

## BS(AI and MMG)
## Spring-2025

**OBJECT ORIENTED PROGRAMMING**

| | |
|---|---|
| Course Title: | Object Oriented Programming |
| Course Code: | CS1-02 |
| Credit Hours: | (3+1) |
| Course Instructor: | Abdul Ghafoor |
| Electronic mail: | aghafoor.faculty@aror.edu.pk |

**Description:**

This course is designed to provide students base for big programming to be able to develop the complex applications inmanageable way that will be more organized, easy to understand, and easy to develop and for stated purpose object-oriented paradigm will be used and for that java as a language is selected. Java is widely used language nowadays and according to oracle more than 10 billion devices are using java in differentapplication domains. This course is designed to target console and desktop (GUI based) applications majorly, other thanthat all the constructs that will be discussed in this course will benefit students for development of java based web applications and the same java based libraries are used for development of mobile applications for well-known mobile operating system android.

**Aims and Objectives:**

- Develop a deep conceptual understanding of the principles and concepts underlying Object-Oriented Programming (OOP).
- Encourage critical thinking and analysis in designing solutions through object-oriented approaches.
- Cultivate problem-solving skills by applying OOP principles to real-world scenarios.
- Enable students to design and implement modular, reusable, and maintainable software systems using OOP principles.
- Instill a mindset of continuous learning by introducing students to emerging trends and advancements in object-oriented programming.
- Encourage students to explore advanced topics in OOP beyond the scope of the course for self-improvement.

## Assessment:

| S. No | **Assessment Activities** | Percentage | Total Activities |
|---|---|---|---|
| 1. | Sessional: Quizzes/ Assignments (Quizzes & Assignments) | 30% | 10 |
| 2. | Mid Term Exam | 30% | 1 |
| 3. | Final Exam | 40% | 1 |

## Course Learning Outcomes (CLOs):

| No. | **Course Learning Outcome** | Domain | Level | Assessment Tool |
|---|---|---|---|---|
| C1 | Demonstrate concepts of object oriented programming paradigm | C | 2 | Class Participation, Quizzes, Mid Exams., Assignments |
| C2 | Demonstrate concepts of basic constructs of java | C | 3 | Class Activity, Quiz, Assignments |
| C3 | Develop console or graphical user interface based applications | C | 3 | Worksheets, Project |

**Domains:**
C=Cognitive, A=Affective, P=Psychomotor
**Levels:**
Cognitive = {1: Remembering, 2: Understanding, 3: Applying, 4: Analyzing, 5: Evaluating,
5: Creating}
Affective = {1: Receiving, 2: Responding, 3: Valuing, 4: Organizing, 5: Characterizing}
Psychomotor= {1: Imitation, 2: Manipulation, 3: Precision, 4: Articulation, 5: Naturalization}

**Course Outlines:**

| Weeks | LEC# | SUBTOPICS . | REFERENCE | Course% Covered |
|---|---|---|---|---|
| Week No: 01 | Lec: 01 | ▪ Introduction to subject<br>▪ History of programming<br>▪ Object oriented programming<br>▪ Java and its history<br>▪ JDK, JRE and JVM<br>▪ Development tools<br>▪ OOP principles<br>▪ First short example<br>▪ Lexical issues<br>▪ Java keywords | Chapter#01 | 2.08% |
| | Lec: 02 | **Java Basic Language constructs:**<br>**1. Data types**<br>**a. Primitive types:**<br>• Integers<br>• Floating points<br>• Characters<br>• Boolean<br>• Integer literals<br>• Floating literals<br>• Character literals<br>• Boolean literals<br>**b. Reference types**<br>• Strings<br>• Arrays<br>• User defined types | Chapter#01 | 4.16% |
| | Lec: 03 | **2. Variables:**<br>• Variables Declaration<br>• Variable Initialization<br>• Scope and lifetime of variables<br>• Type conversion and casting<br>• Explicit casting<br>• Implicit casting | Chapter#01 | 6.25% |

| | | | | |
|---|---|---|---|---|
| | | • Automatic type promotions | | |
| Week No: 02 | Lec 04 | **3. Arrays:**<br>• One dimensional array<br>• Multi-dimensional array | **Chapter#02** | **8.33%** |
| | Lec: 05 | **4. Operators:**<br>• Arithmetic operators<br>• Assignment and compound assignment operators<br>• Bitwise operators<br>• Bitwise logical operators<br>• Boolean logical operators<br>• Relational operators<br>• Short-circuit logical operators<br>• Ternary or three way or ? operator<br>• Operator precedence | **Chapter#02** | **10.41%** |
| | Lec: 06 | **5. Control Statements:**<br>**a. Selection Statements**<br>• If<br>• Nested Ifs<br>• If-else-if ladder<br>• Switch statement<br>• Nested switch statements | **Chapter#02** | **12.49%** |
| Week No: 03 | Lec: 07,08 | **b. Iteration Statements:**<br>• for<br>• while<br>• Do-while<br>• for each<br>• Nested Loops | **Chapter#02** | **16.65%** |
| | Lec: 09 | **c. Jump Statements:**<br>• Break<br>• Continue<br>• Return<br>• Throw | **Chapter#02** | **18.73%** |
| W e e . | Lec: 10 | **6. Introducing classes**<br>• General form of a class | **Chapter#02** | **20.81%** |

| | | | | |
|---|---|---|---|---|
| | | • Declaring objects<br>• Assigning object reference variables or initializing objects | | |
| | **Lec: 11** | • Methods<br>• Constructors<br>• Parameterized Constructor<br>• This keyword<br>• Resolving Instance variable hiding | **Chapter#02** | **22.89%** |
| | **Lec: 12** | • Overloading methods<br>• Overloading constructors<br>• Parameters and arguments<br>• Pass by value<br>• Pass by reference<br>• Objects as parameters and arguments | **Chapter#03** | **24.97%** |
| **Week No:05** | **Lec: 13** | • Returning objects<br>• Setter and getter methods or Accessors and mutators<br>• Recursion<br>• Access control<br>• Public<br>• Protected<br>• Private | **Chapter#03** | **27.05%** |
| | **Lec: 14,15** | • Static keyword<br>• Static with classes<br>• Static with methods<br>• Static with fields<br>• Static blocks<br>• Static with import keyword<br>• Final keyword<br>• Nested or inner classes | **Chapter#03** | **31.21%** |

| | | | | |
|---|---|---|---|---|
| Week No:06 | Lec: 16 | • Type wrappers<br>• String handling<br>• Random class<br>• Math class<br>• Object class | Chapter#03 | 33.29% |
| | Lec: 17 | **7. Inheritance:**<br>• Member access and inheritance<br>• Super keyword<br>• Super for accessing parent class members<br>• Super for invoking parent class constructors<br>• Multi-level class hierarchy and constructor execution means constructor hierarchy<br>• Method overriding<br>• Dynamic method dispatch | Chapter#03 | 35.37% |
| | Lec: 18 | **8. Abstraction:**<br>• Abstract classes<br>• Abstract methods<br>• Concrete methods | Chapter#03 | 37.45% |
| Week No:07 | Lec: 19 | • Interfaces<br>• Interface definition`<br>• Interface implementation<br>• Accessing implementations through interface references<br>• Nested interfaces<br>• Applying interfaces | Chapter#03 | 39.53% |
| | Lec: 20,21 | • Variables in interfaces<br>• Interfaces extension<br>• Default interface methods<br>• Default method fundamentals<br>• Multiple inheritance | Chapter#03 | 41.61% |

| | | | | |
|---|---|---|---|---|
| Week No: 08 | Lec: 22 | **9. Exception handling:**<br>• Exception handing fundamentals<br>• Exception types<br>• Exception<br>• Error | **Chapter#04** | **44.71%** |
| | Lec: 23 | • Checked and Unchecked exceptions<br>• Uncaught exceptions | **Chapter#04** | **46.81%** |
| | Lec: 24 | • Using try and catch<br>• Displaying description of exception | **Chapter#04** | **49.91%** |
| Week No: 09 | Lec: 25 | • Multiple catch classes<br>• Nested try statements<br>• Throw keyword | **Chapter#04** | **52.00%** |
| | Lec: 26 | • Throws keyword<br>• Finally block created with finally clause/keyword<br>• Built in Exceptions | **Chapter#05** | **54.09%** |
| | Lec: 29 | ○ **Summing Up Exception Handling Concepts** | **Chapter#07** | **56.18%** |
| Week No: 10 | Lec: 30 | **10. Filing:**<br>• Opening a file<br>• Reading From a file<br>• Writing to a file<br>• Closing a file | **Chapter#07** | **58.27%** |
| | Lec: 31 | **11. GUI in Java:**<br>a. Introduction to AWT<br>b. Introduction to Event Handling | **Chapter#07** | **60.36%** |
| | Lec: 32 | c. Introduction to Swing | **Chapter#07** | **62.45%** |
| Week No: 11 | Lec: 33 | • Working with windows<br>• Swing controls<br>• Components | **Chapter#07** | **64.54%** |

| | | | | |
|---|---|---|---|---|
| | **Lec: 34,35** | • JFrame<br>• JPanel<br>• JButton, JLabel | **Chapter#07** | **70.34%** |
| **Week No: 12** | **Lec: 36** | • JManeuBar, JItems<br>• JDropDown<br>• Swing Utility Functions (i.e size, color, disable, etc ) | **Chapter#10** | **72.42%** |
| | **Lec: 37** | • Summing Up GUI Libraries | **Chapter#10** | **74.5%** |
| | **Lec: 38** | • Some famous GUI Application Examples | **Chapter#10** | **76.58%** |
| **Week No:13** | **Lec: 39** | **12. JDBC:**<br>• Connecting to the Database | **Javatpoint, GeeksforGeeks** | **78.66%** |
| | **Lec: 40** | • CRUD Operations on the database | **Javatpoint, GeeksforGeeks** | **80.74%** |
| | **Lec: 41** | • Store a file<br>• Retrieve a file<br>• Store and Retrieve an Image | **javatpoint, GeeksforGeeks** | **82.82%** |

| | | | | |
|---|---|---|---|---|
| **Week No: 14** | **Lec: 42** | **13. Collection Framework:**<br>• Array List and Linked List | **Javatpoint, GeeksforGeeks** | **84.9%** |
| | **Lec: 43** | **14. Servlet and JSP:**<br>• Introduction to servlet<br>• Creating a Dynamic web project using servlet | **Javatpoint, GeeksforGeeks** | **86.98%** |
| | **Lec: 44** | • Deployment Descriptor<br>• Servlet Annotations<br>• Http Servlet Request and Response | **Javatpoint, GeeksforGeeks** | **90%** |

| | | | | |
|---|---|---|---|---|
| **Week No:15** | **Lec: 45** | • doGet vs doPost Method<br>• sendRedirect Method<br>• Http Session Management in Servlet | **Javatpoint** | **93%** |
| | **Lec: 46,47** | • Introduction to JSP<br>• JSP Tags, objects, Expression Language<br>• Login using: Servlet+JSP+JDBC | **Javatpoint** | **100%** |
| **Week No: 16** | **REVISION** | | | |

## Text Books:

1. Java the Complete Reference by Herbet schilit
2. Absolute Java, 6th edition by Walter Savicth

## Reference

**Web Tutorials:**

- https://www.geeksforgeeks.org/c-plus-plus/
- https://www.javatpoint.com