

University of Asia Pacific (UAP)
Department of Computer Science and Engineering (CSE)
BSc in CSE Program

Course Outline – Object-Oriented Programming I Lab

Part A – Introduction

1. **Course No. / Course Code:** CSE 202
2. **Course Title:** Object-Oriented Programming I Lab
3. **Course Type:** Core course
4. **Level/Term and Section:** 3rd Semester (2nd Year/1st Semester)
5. **Academic Session:** Fall 24
6. **Course Instructor:** Tanjina Helaly, Baivab Das
7. **Prerequisite (If any):** None
8. **Credit Value:** 1.5
9. **Contact Hours:** 3.0
10. **Total Marks:** 100
11. **Course Objectives and Course Summary:**

The objectives of this course are to:

1. **Teach** OOP principles and features and how to apply them in solving real life problem using java.
2. **Demonstrate** how to use a modern IDE to develop java application.
3. **Show** how to use Java library effectively.

This course will cover the main aspects of the Java programming language. Students will learn how to use Java according to proper Object-Oriented Programming principles. This course covers the Java language syntax, and then moves into the object-oriented features of the language. Students will then learn the OOP principles, Data types, Variables, Scoping and life time of variable, Operators, classes and objects, Inheritance, Abstraction, Exception Handling, Threading, File and StringTokenizer, Networking, I/O streams, Collections API packages

12. Course Learning Outcomes: at the end of the Course, the Student will be able to –

CLO 1	Develop applications using programming language basics.
CLO 2	Develop well-designed applications using the OOP features.
CLO 3	Use a modern/popular IDE to develop the java application.
CLO 4	Use the Library effectively.
CLO 5	Independently learn and apply new knowledge or techniques to solve a problem.

13. Mapping / Alignment of CLOs with Program Learning Outcomes (PLO) (Optional):

CLO No.	Corresponding PLOs (Appendix-1)	Bloom's taxonomy domain/level (Appendix-2)	Delivery methods and activities	Assessment Tools
CLO1	3	1/Apply	Lecture, Problem Solving	Class work, Problem-solving, Exam
CLO2	3	1/Analyze	Lecture, Problem Solving	Class work, Problem-solving, Exam, Project
CLO3	5	2/Manipulate	Lecture, Demonstration	Class work, Project
CLO4	3	1/Apply	Lecture, Problem Solving	Class work, Problem-solving, Exam, Project
CLO5	12	1/Analyze	Lecture, Problem Solving	Class work, Project

CLO5:

- Independently find and interpret discipline related documentation.
- Able to analyze a given program and able to debug, extend, improve application behavior according to given instructions.
- Able to search and use the program library for some standard objects

Part B – Content of the Course

14. Course Content:

15. Alignment of topics of the courses with CLOs:

SL. No	Topics / Content	Course Learning Outcome (CLO)
1	Language Basic	CLO1, CLO3
2	OOP Basics	CLO2, CLO3
3	OOP Principles: Inheritance, Encapsulation, Polymorphism, Abstraction	CLO2, CLO3, CLO5
4.	Language Specific library (Exception, Thread, IO, Collections)	CLO3, CLO4, CLO5

16. Class Schedule/Lesson Plan/Weekly plan:

Topics	Specific Outcome(s)	Time Frame	Suggested Activities	Teaching Strategy(s)	Alignment with CLO
Introduction to Java, Object-Oriented Programming., Hello World program	Understand the Java development environment • Write and execute a basic Java program	Week 1	Write "Hello World", identify Java syntax elements	Lecture, multimedia, demonstration	CLO1
Tool Set up, Develop simple application covering Java basics and User input.	Set up JDK and IDE (e.g., IntelliJ, Eclipse) • Develop Java programs using variables, input, and basic output	Week 2	Lab setup, Develop simple program (Problem-solving)	Lecture, multimedia, demonstration	CLO1
Develop Simple application involving creating Object and accessing members of the class.	• Define classes and instantiate objects • Access fields and methods of a class	Week 3	Create and test user-defined class with fields/methods	Lecture, multimedia, demonstration	CLO2, CLO3, CLO4
Continuation of Class and Object to make students get a good grasp in OOP	• Strengthen understanding of class structure and	Week 4	Enhance previous programs, introduce	Lecture, Problem Solving,	CLO2, CLO3, CLO4

basic concept class and Object.	object use • Apply OOP principles in small programs		interactive program (Menu)	Group discussion	
Introduce Overloading. Develop a simple management system.	Implement method overloading • Design modular structure of a management system	Week 5	Create overloaded constructors in a system like library or student manager	Lecture, multimedia, demonstration	CLO2, CLO3, CLO4
Introduce Inheritance and Overriding and implement those features in the management system.	• Use inheritance to extend classes • Apply method overriding in class hierarchy	Week 6	Refactor system with superclass and subclasses using inheritance	Lecture, multimedia, demonstration	CLO2, CLO3, CLO4
Implement Abstraction and Subclass Polymorphism in the management system	• Apply abstraction using interfaces and abstract classes • Implement polymorphism via subclassing	Week 7	Implement abstract base class/interface in the management system	Lecture, multimedia, demonstration	CLO2, CLO3, CLO4
Mid Exam – Provide the problem statement of a management system and Students have to design the classes and implement the system. (Week 8)					
Introduce Exception Handling and implement it into the Management system	• Handle exceptions using try-catch-finally • Design robust applications with exception handling	Week 9	Implement Custom Exception and refactor management system to include exception handling	Lecture, multimedia, demonstration	CLO2, CLO3, CLO4
Implement GUI and Event Handling into the management system. (Java FX)	• Design GUI using JavaFX • Implement basic event handling mechanisms	Week 10	Create forms, buttons, input handlers for management system	Lecture, multimedia, demonstration	CLO2, CLO3, CLO4

Continue GUI	<ul style="list-style-type: none"> • Enhance GUI interaction • Use advanced GUI features (e.g., layout, styling) 	Week 11	Extend previous GUI with additional features like menus, dialogs	Lecture, multimedia, demonstration	CLO2, CLO3, CLO4
Implement Java IO and String manipulation into the management system	<ul style="list-style-type: none"> • Read/write to files using Java IO • Perform operations on strings 	Week 12	Save/load management data, parse inputs using string methods	Lecture, multimedia, demonstration	CLO2, CLO3, CLO4, CLO5
Project Submission	<ul style="list-style-type: none"> • Demonstrate complete understanding by submitting a full application • Document and present code 	Week 13	Final project demo	Lecture, multimedia	CLO2, CLO3, CLO4, CLO5
Final Exam (Week 14)					

17. Teaching-Learning Strategies: Demonstration, Hands-On Lab work, Group Project

18. Assessment Techniques of each topic of the course:

SL. No	Topics / Content	Assessment Techniques
1	Language Basic	Class work, Problem-solving, Exam
2	OOP Basics	Class work, Problem-solving, Exam, Project
3	OOP Principles: Inheritance, Encapsulation, Polymorphism, Abstraction	Class work, Problem-solving, Exam, Project
4.	Language Specific library (Exception, Thread, IO, Collections)	Class work, Problem-solving, Exam, Project

Part C – Assessment and Evaluation

19. Assessment Strategy

Class Performance: Students will be provided with some problems during each class and evaluated based on the number of solved problems and correctness.

Assignment: Unfinished work should be submitted as assignment.

Additional assignments may be given as needed. **Copied** home work will be graded as **zero**. **Late** submission will result a **50% deduction** in score.

Project: Students have to develop an application exercising the OOP features (Inheritance, Polymorphism, Abstraction), Programming language specific libraries (such as Exception, IO, etc.)

Mid Exam (on Problem solving): The problem statement of a small management system will be provided and Students have to design the classes and implement the system.

Final Exam: A test covering the basic knowledge of CLO1-CLO5.

CIE- Continuous Internal Evaluation (55 Marks)

Bloom's Category Marks (out of 30)	Class Performance (25)	Assignments (10)	Project (20)
Remember			
Understand			
Apply	25	10	15
Analyze			5
Evaluate			
Create			

SMEB- Semester Mid & End Examination (45 Marks)

Bloom's Category	Test
Remember	5
Understand	5
Apply	30
Analyze	5

Evaluate	
Create	

Weighting COs with Assessment methods:

Assessment Type	% weight	CLO1	CLO2	CLO3	CLO4	CLO5
Final Exam MCQ/Short question quiz	25%	5	13	3	4	
Mid Term Problem Solving Exam followed by Viva	20%		15	5		
Project followed by viva	20%		10	4	2	4
Class performance	25%	3	13	5	4	
Assignment	10%		10			
Total	100%	8	61	17	10	4

20. Evaluation Policy

Grades will be calculated as per the university grading structure and individual student will be evaluated based on the following criteria with respective weights.

1. Class Performance 25%
2. Assignment 10%
3. Mid-Term Examination 20%
4. Term Examination 25%
5. Project 20%

UAP Grading Policy

Numeric Grade	Letter Grade	Grade Point
80% and above	A+	4.00
75% to less than 80%	A	3.75
70% to less than 75%	A-	3.50

65% to less than 70%	B+	3.25
60% to less than 65%	B	3.00
55% to less than 60%	B-	2.75
50% to less than 55%	C+	2.50
45% to less than 50%	C	2.25
40% to less than 45%	D	2.00
Less than 40%	F	0.00

Part D – Learning Resources

21. Text Book

1. Herbert Schildt and Danny Coward, Java the Complete Reference, 12th edition (2021), McGraw Hill

Reference Books & Materials

1. জাভা প্রোগ্রামিং - আন ম বজলুর রহমান
2. Paul Deitel and Harvey M. Deitel, Java: How to Program, 9th Edition (2011), Pearson College Div
3. Kathy Sierra and Bert Bates, Head First Java, 2nd Edition (2005), O'Reilly Media
4. Others