# LAB SESSION 8

**Open Ended Lab**

**Objective**

The objective of this lab session is to assess your understanding and application of Object-Oriented Programming (OOP) concepts, problem-solving skills, and coding techniques. You will be evaluated based on your ability to design and implement a system using OOP principles within the session.

**Introduction**

In this lab session, you will be presented with a general problem that you need to solve. Your solution will be assessed based on the following criteria: technique, problem-solving skills, OOP concepts, code efficiency, and code readability. Ensure that you apply the knowledge and skills covered in previous labs to complete the task effectively.

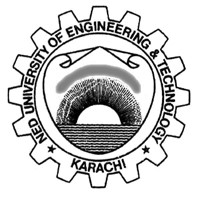
**Problem Statement**

You will need to design and implement a system using OOP principles.

Each criterion will be scored as follows:

* **0**: The criterion is not met.
* **1**: The criterion is met.

Please review your previous lab notes and understand the key concepts of OOP, including class design, inheritance, encapsulation, and polymorphism, before attempting this assessment.



**NED University of Engineering & Technology**

**Department of Software Engineering**

**Object Oriented Concepts and Programming**

|  |  |  |
| --- | --- | --- |
| **OEL ASSESSMENT** | | |
| **SKILL SETS** | **EXTENT OF ACHIEVEMENT** | |
| **CRITERIA** | **0** | **1** |
| **Technique**  Applies programming techniques and methodologies. | Fails to apply programming techniques and methodologies. | Correctly applies programming techniques and methodologies. |
| **Problem-Solving Skills**  Solves the given problem using appropriate logic and methods. | Fails to solve the given problem using appropriate logic and methods. | Effectively solves the given problem using appropriate logic and methods. |
| **OOP Concepts**  Demonstrates understanding of OOP concepts (e.g., inheritance, encapsulation, and polymorphism). | Fails to Demonstrate a solid understanding of OOP concepts (e.g., inheritance, encapsulation, and polymorphism). | Demonstrates a solid understanding of OOP concepts (e.g., inheritance, encapsulation, and polymorphism). |
| **Code Efficiency**  Writes optimized code. | Fails to write efficient and optimized code. | Writes efficient and optimized code. |
| **Code Readability**  Ensures the code is well-organized and easy to read and understand. | Fails to Ensure that the code is well-organized and easy to read and understand. | Ensures the code is well-organized and easy to read and understand. |

|  |  |
| --- | --- |
| **Weighted CLO (Psychomotor Score)** |  |
| **Remarks** |  |
| **Instructor’s Signature with Date:** |  |