Faculty of Computing

CS110: Fundamentals of Computer Programming

Class: BESE-16B

Lab 08: Open Ended Lab

CLO 2	**Solve** given real-world problem by applying appropriate programming concepts and techniques
CLO 3	**Build** a program and associated documentation using appropriate IDE and supplementary tools.
CLO 4	**Perform** effectively as an individual or member of a team.

Date: 28th October 2025

Time: 2:00pm-5:00pm

Instructor: Dr. Momina Moetesum
Lab Engineer: Mr. Nadeem Nawaz

Task:
The objective of this open-ended lab is to allow students to explore, design, and develop a **text-based detective game** using the relevant programming constructs covered so far. The game should simulate a detective investigation where the player explores different locations, interacts with suspects, collects clues, and ultimately solves the case by identifying the real culprit.

The application should have the following characteristics:

- There should be a proper **game logic** with **win and lose conditions**, such as correctly solving the case or failing due to wrong accusations or depleted energy.
- Ensure **player navigation** through the game by applying **text-based interactions**, allowing the detective to move between locations (e.g., "Study Room", "Garden", "Kitchen").
- There should be **challenges and clues** that influence the player's progress. For example, gaining credibility from correct deductions or losing energy for wrong decisions.
- There should be an **element of randomness** in the game, such as unpredictable suspect behavior, random availability of clues, or chance-based outcomes.
- There should be **multiple stages of investigation**, where the complexity or difficulty increases as the player progresses (e.g., initial clue-gathering, suspect interrogation, final accusation).
- There should be a simple **progress-tracking mechanism** using variables such as energy, score, or reputation to represent the player's status.
- The game should include an **appealing and user-friendly text interface** that engages the player through descriptive messages, narrative prompts, and clear formatting (a graphical interface is optional).

To attempt this lab students must:

1. Explore various existing text-based games.
2. Explore various options to enhance user experience.
3. Employ good programming practices like standard naming convention, modularity, comments and error handling.

Submission Requirements:

This is a group task where each group must comprise of 2-3 members. There should be a clear distribution of tasks to ensure teamwork. The submission time for this task is till next week's lab. The deliverables of this task include:

- A brief report describing the design and methodology of your game. Also explain the game logic. Use flow charts to explain your game's working. Include screenshots of various stages.
- Code files with ReadMe.txt for instructions to play your game.
- Demonstrate the working of your game in the next lab.

Lab Rubrics:

Your Lab 8 will be graded out of 5 for each rubric according to the following rubrics.

Lab Rubrics for Lab 8					
Sr. No.	**Assessment**	**Unacceptable (0 Marks)**	**Does Not Meet Expectations (1/2 Marks)**	**Meets Expectations (3/4 Marks)**	**Exceeds Expectations (5 Marks)**
1	**Application of Programming Concepts (CLO2, PLO3)**	The student did not submit any work. OR The student plagiarized the solution and/or used unfair means.	The student is unable to apply the appropriate programming concepts to solve the given problem thus resulting in an incomplete or ineffective solution. The program flow is messy and incomprehensible. Codes are non-modular and cannot be reused.	The student requires some guidance to apply the appropriate programming concepts to solve the given problem. The program flow requires minor improvements. Codes are semi-modular and semi-reusable.	The student demonstrates a clear ability to apply the appropriate programming concepts to solve the given problem. The program flow is adequate. Codes are modular, reusable, and easily readable.
2	**Software Tool Usage (CLO3-PLO5)**		The student demonstrates a lack of understanding of tool usage. Implementation has syntax/semantic/runtime errors, and the student is unable to debug and correct the errors. The code has inadequate comments and variable names and does not adhere to the coding standards. No Error handling has been performed. Documentation is poorly structured.	The student demonstrates some understanding of tool usage. The codes are correct in terms of their syntax, however, the program output is not always correct in all test cases. The code has limited comments and inconsistent variable names and may not adhere to the coding standards. Some Error handling has been performed. Documentation is adequately structured.	The student demonstrates a good understanding of tool usage. Furthermore, his/her coding is complete and functional, and the program output is correct in all test cases. The code has sufficient comments and consistent variable names and reasonably adhere to the coding standards. Adequate Error handling has been performed. Documentation is well structured.
3	**Individual and Teamwork (CLO4, PLO9)**		The student is unable to cooperate in a group setting. Demonstrates forced cooperation through intervention.	The student is able to cooperate in a group setting. Reluctant to take a leadership role or lacks personal dominance.	The student actively engages and collaborates with group members in an effective manner. The student not only collaborates but also takes up leadership roles.