The German International University
Faculty of Informatics and Computer Science
Dr. Nada Sharaf

Programming III, Winter Semester 2024/2025
Milestone 01

Deadline 22nd of November 11:59 PM

General Description:
In this project, you are tasked with implementing a simple 2D card matching game that can be played via console application. The game is based on the classic concept of memory matching, where players flip over cards to find pairs. The game continues until all pairs have been matched, with each player trying to identify pairs based on memory. The player with the most pairs at the end of the game wins.

Milestone 01: In this milestone, you are asked to implement the following classes and functions, each class should have appropriate member variables and member functions:

a) **Card Class:** The Card class represents an individual card in the game. Card class will serve as the base class for the special cards which are Standard, Bonus and Penalty cards.

Methods:

- display(): Displays the card's number if face-up or a placeholder (*)if face-down.

Card Subclasses:

- **StandardCard:** This represents a regular card with no special abilities.
- **BonusCard:** This card offers bonus points when matched.
- **PenaltyCard:** This card deducts points if mismatched.

b) **Deck Class:** The Deck class represents the collection of cards in the game. It should manage the array of Card objects and include the following methods:

Methods:

- shuffle(): Randomly shuffles the deck of cards.
- displayGrid(): Displays the current layout of the cards in the grid. For current milestone the 4x4 grid should look something like that:

*	*	*	*
*	*	*	*
*	*	*	*
*	*	*	*

c) **Player Class:** The Player class represents an individual participant in the game. It includes details such as the player's name and score

Methods:

- displayScore(): Displays the player's current score.

d) **Game Class:** The Game class manages the overall game flow and player interactions.

Methods:

- initializeGame(): Initializes the game, shuffles the deck, and places cards to the grid.

Milestone Requirements: This milestone focuses on the structure of the game. Full game-play functionalities (like turns, scoring, etc.) will be implemented in milestone 02.

0.1 Deliverables:

- A Header file for each of the previously mentioned classes.
- **One single Source file (.cpp)** with the implementation of the previously mentioned classes, methods and attributes.
- The source file should have a main method that must initialize the game, shuffle the deck, and display the grid for testing purposes.
- All those files should be then put in one folder and the folder should be compressed into a .zip file with following naming format.`TeamNumber_TheNameOfTheSubmittingMember_theIDofTheSubmittingMember.`
- PLEASE DO NOT EXTRACT A FULL PROJECT AND JUST SUBMIT THE FILES AS MENTIONED ABOVE

Important notes:

- **CHEATING CASES WILL LEAD TO 0 IN THE PROJECT**
- **FULLY AI GENERATED CODE WILL LEAD TO A 0 IN THE PROJECT**
- All attributes in the classes are private and has setters and getters that has to be implemented
- Each class should have 2 constructors: the default constructor and a constructor that initialize all the attributes of the classes
- Each class should have a destructor even if it will remain empty
- The Grid array and the cards attributes should be dynamically created
- All the functionalities and game play (turns, deals and scoring, etc..) as well as the special behavior of each of the subclasses will be discussed and implemented in milestone 02.