Python for Computer Science and Data Science 2 (CSE 3652)
MAJOR ASSIGNMENT-1: OBJECT-ORIENTED PROGRAMMING (OOP)

Card Game Simulation and Analysis

1 Problem Statement

Design and implement a basic card game framework that supports simple card games like Blackjack. The framework should include a deck of cards, players, and basic game rules. The goal of this project is to create an extendable and reusable structure that can accommodate the card game by implementing fundamental components such as a deck of cards, a hand, and a set of game rules.

For example, in Blackjack, the objective is to achieve a hand value as close to 21 as possible without exceeding it. The game involves a dealer and one or more players. The dealer follows fixed rules for drawing cards, while players can decide whether to draw additional cards.

2 Tasks

2.1 Task 1: Create Card and Deck Classes

- Create a `Card` class that represents a single playing card. Each card should have a **suit** (Hearts, Diamonds, Clubs, or Spades) and a **rank** (Ace, 2, 3, ..., 10, Jack, Queen, King).

- Use an `enum` type to define suits instead of using string values. This ensures consistency and prevents errors due to typos.

- Implement a `Deck` class that generates a full deck of 52 cards, consisting of four suits and thirteen ranks each.

- Add functionality to shuffle the deck randomly.

- Implement a method to draw a card from the deck, removing it from the available cards.

- **Example Data:**

  - Example of a single card: `Card(suit=Hearts, rank=Ace)`.
  - Example of a shuffled deck: `[Card(Spades, King), Card(Diamonds, 7), Card(Hearts, 2), ...]`.

2.2 Task 2: Implement a Simple Blackjack Game

- Blackjack is a game where players aim to get as close to 21 points as possible without exceeding it. Face cards (King, Queen, Jack) are worth 10 points, Aces can be worth 1 or 11 points, and other cards retain their numeric values.

- The dealer and the player are both dealt two cards at the start of the game.

- The player can choose to **hit** (draw an additional card) or **stand** (don't take any more cards.).

- If the player's total exceeds 21, they go **bust** and lose the game.

- The dealer must follow a set rule: if their total is 16 or below, they must hit; if 17 or above, they must stand.

- The player wins if their total is higher than the dealer's without exceeding 21.

- **Example Scenario:**

  - **Player's Hand:** `[Card(Hearts, 10), Card(Clubs, 7)]`
    Total: 17
  - **Dealer's Hand:** `[Card(Spades, 8), Card(Diamonds, 10)]`
    Total: 18
  - **Result:** Dealer wins.

2.3 Task 3: Implement Multiplayer Support

- Extend the framework to support multiple players in a game.

- Allow each player to take turns making decisions (e.g., hit or stand in Blackjack).

- Implement a turn-based system to manage player actions and game progression.

- Display player hands and game status after each round.

- **Example Data:**

  - Player 1 hand: `[Card(Hearts, 7), Card(Spades, 5)]`.
  - Player 2 hand: `[Card(Diamonds, 9), Card(Clubs, 6)]`.
  - Player 1 hits and receives `Card(Hearts, 3)`.
  - Updated hand: `[Card(Hearts, 7), Card(Spades, 5), Card(Hearts, 3)]` (Total = 15).

NB:

Basic Rules of the Game:

- The goal is to get as close to 21 as possible without going over.

- Face cards (King, Queen, Jack) = **10 points**.

- Aces (A) = **1 or 11 points**, depending on what benefits the player.

- If your total is higher than the dealer's without busting (going over 21), **you win**.

- If you go over 21, you automatically **lose (bust)**.

- The dealer must **hit if they have 16 or less** and **stand on 17 or more**.

When Does a Blackjack Game End?

A game of Blackjack concludes in the following scenarios:

1. **Immediate Loss: Player Busts**

   - If a player's total exceeds **21**, they automatically lose, and their participation in the game ends.
   - Other players continue until they either bust or stand.

2. **Immediate Win: Dealer Busts**

   - If the dealer's total exceeds **21**, all remaining players who have not busted automatically **win**.
   - The game ends at this point.

3. **All Players Have Stood**

   - Once every player has either **busted** or **stood**, the dealer plays their turn.
   - The dealer must:
       - **Hit on 16 or less**.
       - **Stand on 17 or more**.
   - The game proceeds to the final comparison when the dealer stands or busts.

4. **Final Comparison (Game Ends)**

   - The dealer's total is compared with each standing player:
       - If the **player's total is higher than the dealer's**, the player **wins**.
       - If the **dealer's total is higher**, the player **loses**.
       - If both have the **same total**, it's a **push (tie)**, and the player's bet is returned.
   - After this comparison, the game ends.