

Project Part Two: Blackjack Players and Hands of Cards

Implementing Blackjack Players

For the game of Blackjack, we will need classes which represent each player at the Blackjack table and the dealer. We will also need a class to represent the cards which have been dealt to the players and the dealer.

During a game of Blackjack each player (including the dealer) are dealt a set of playing cards called a “hand”. Each card has a point value. Cards whose face value is two through nine have the corresponding point value. Cards whose face value is ten, jack, queen or king have a point value of 10. Aces are a special case: they have a point value of 1 or 11 whichever is better for the player. The player’s current score is the sum of the point values of the cards which the player possesses. Having more than 21 points is bad for the player (or dealer) and is called “going bust” or “busted”. In this case the player (or dealer) loses.

For this part of the project, we are not yet concerned with the complete game of Blackjack. We are creating the classes to represent the player, the dealer and their corresponding collections of cards.

Requirements for the Hand of Cards class

The following diagram shows the necessary methods for the Hand of Cards class:

| Hand of Cards |
|--------------------------------|
| HandOfCards(); |
| void addCard(Card someCard); |
| void removeAllCards(); |
| void display(bool faceDown); |
| int getScore(); |

- This class should maintain a list of individual playing cards.
- The default constructor (HandOfCards) creates an empty list of cards.
- The addCard method adds one new card to the list.
- The removeAllCards method resets the list so that it does not contain any cards.
- The display method will display each card currently in the list. The boolean parameter, faceDown, determines whether the first card (and only the first) in the list is displayed face up or face down.
- The getScore method will calculate the best Blackjack score possible for the cards in the list. The best Blackjack score is the highest possible score less than 21. For instance a hand consisting of a nine and ace should be score as 9 + 11 not 9 + 1.

Requirements for the Player class

The following diagram shows the necessary methods for the Player class:

| Player |
|--------------------------------|
| Player(); |
| void addCard(Card someCard); |
| void removeAllCards(); |
| void display(); |
| int getScore(); |
| void playHand(); |
| bool wantsAnotherCard(); |
| bool wantsToStand(); |
| bool hasBusted(); |

- The Player class should maintain a hand of playing cards.
- The default constructor creates an empty hand of cards.
- The addCard method adds one new card to the hand of cards.
- The removeAllCards method resets the hand of cards so that it is empty.
- The display method will display each card currently in the list. For the Player class, display, will always display the cards face up.
- The getScore method will calculate the best Blackjack score possible for the Player.
- The playHand method will ask the player (by getting user input using cin) whether they want another card (“hit”) or whether they want to stop accepting more cards (“stand”). This method should then set data member variables so that wantsAnotherCard() and wantsToStand() perform correctly.
- The wantsAnotherCard method will return true if the Player wants another card.
- The wantsToStand method will return true if the Player does not want anymore cards.
- The hasBusted method will return true if the Player’s score is more than 21.

Requirements for the Dealer class

The following diagram shows the necessary methods for the Dealer class (note that this is very similar to the Player class):

| Dealer |
|--------------------------------|
| Dealer(); |
| void addCard(Card someCard); |
| void removeAllCards(); |
| void display(bool faceDown); |
| int getScore(); |
| void playHand(); |
| bool wantsAnotherCard(); |
| bool wantsToStand(); |
| bool hasBusted(); |

- The Dealer class should maintain a hand of playing cards.
- The default constructor creates an empty hand of cards.
- The addCard method adds one new card to the hand of cards.
- The removeAllCards method resets the hand of cards so that it is empty.
- The display method will display each card currently in the list. For the Dealer class, a boolean parameter will determine whether the first card is displayed face up or face down.
- The getScore method will calculate the best Blackjack score possible for the Dealer.
- The playHand method will decide whether the Dealer wants more cards or not. The Dealer wants another card if the Dealer's current score is less than or equal to 16. The Dealer wants to stand if the Dealer's current score is more than 16.
- The wantsAnotherCard method will return true if the Dealer wants another card.
- The wantsToStand method will return true if the Dealer does not want anymore cards.
- The hasBusted method will return true if the Dealer's score is more than 21.