16/20

Blackjack Program

Abstract:

The code that I am writing is a Blackjack program. The blackjack program will allow the user to play against a AI computer. The user will be able to play the computer and a score will be kept track of how many times the user beats the computer simultaneously against Blackjack. At the end after the user loses they will be prompted to enter a high score in a table and it will be displayed after the user enters their high score. If a user has already played their old high score will be replaced with the new one if the new one is higher than the old one. The program will require me to make lists, arrays, many methods as well as print formatting statements and such.

Introduction:

I was having a real hard time choosing a project idea. I wanted to choose something that would be practical in real life, should be challenging but at the same time would be realistic. I finally settled on choosing Blackjack because it is something that I think is interesting and also fun. Blackjack is a relatively simple game to teach so I decided that it would be something that could be interesting to play. At the same time, casinos are extremely popular because adults love gambling as we move into the future, I believe that maybe online casinos will be a thing, Therefore writing a code for blackjack might be something that might be useful down the line. Potentially I could sell my code to a casino if it was good enough and could make money off of it making this code practical.

Detailed System Description:

The system allows the user to play Blackjack and interaction with the simple will be quite simple. The user is given 2 random cards from a deck, those two cards are taken out of the deck. Afterwards the user is prompted whether they want to add another card to the cards they currently have or just stay with the cards that they have. Each card will have a value anywhere from 1 to 11. A sum will be kept track of the user's card values. If at any point the user's card values add up to a sum over 21 they bust or lose out instantly unless the AI also busts. Speaking of the AI, the AI also will get two cards which will also be taken out of the deck and will also be programmed to hit or stay based on what their current sum of their card values is. Each time the user or AI hits the card will be taken out of the deck to keep the game as realistic as possible. All the user will need to do is hit or stay, whenever the user wants to stay or busts the game will end and the user's card values and the AI's card values will be compared.

Whoever has a higher sum closer to 21 wins that round, if the sum is the same then the user and AI draw and the round is replayed. Also the user loses to the AI they will be prompted to enter their name into a high score table that will be displayed. If they already have high score, if they got a new higher one the old one will be replaced.

Requirements:

The problem that the report is addressing is the game of Blackjack, not much to it other than that.

Literature Survey:

I know for sure that other blackjack game variants are available out there on the web. Most of them won't be similar to mine because they are based off of interactive user interfaces and not command line code. I will also try to attempt to tackle user interfaces but I don' know if I will be able to implement that by the end of the semester.

User Manual:

The system should be used to play Blackjack. To use the system the user will be entering the following words: "hit", "stay" or their name and based on what the user enters different aspects of the program will run.

Conclusion:

The goals of this project will allow the user to play a game of Blackjack and if I am able to implement a user interface, then the goals will be to allow the user to interactively play a game of blackjack. I hope to make this interactive because then I can change the code so that many players can play at once and make many other modifications to make the program work in neat ways.

References/Bibliography:

Stack overflow: lists, arrays

What is left:

I have the deck already implemented and it tracks the user's cards. It also allows the user to hit a single time and it tracks the user's current cards and their sum. After the user hits it will give them another card and increase the sum of the user's card values. I have not yet implemented a stay function but that is easy, I just need to break out of the look and end the program. I have also not worked on the AI yet but I have experimented with how it would work and have some ideas. Lastly, I have not created a board of high scores but I think I might know how to do that as well

UML Diagram:

Casino1
-Blackjack(): void
-main(String[]) : void

This is part of your detailed system description

You could have a better design for this. For example, you could have classes for Card, Deck, etc. and methods to evaluate a hand, etc.