Sally Anderson 9/10/2021

BlackJack – Portfolio Project

1. Introduction

- This is a text based single player version of the classic card game Blackjack.
- The player tries to beat the computer at obtaining 21 without going over. The closest one wins. The dealer and player are each dealt two cards. One of the computer cards is dealt face down. The player is asked if he wants another card or will stay where he is. The value of his hand is totaled and displayed. If the player doesn't go bust, the computer then tries to get 21. At 17, the computer will stand where he is. The winner is determined.

2. Design and Implementation

- This version of Blackjack is designed with 5 classes, the card, the deck, the player, the dealer, and the game.
    - Card - represents the individual cards, suit and value
    - Deck – represents a deck of cards, (4 suits and card values Ace through King). Shuffle and draw functions
    - Player – Hand, total value of cards, number of aces. Add a card and score functions
    - Dealer – inherited from Player class same as player
    - Game – controls the flow of the game. Play, hit, and stand functions
- Conversion from written description to coded functions to code with classes
    - This project was originally written without classes. In the conversion process to classes, I designed the classes and what attributes I would need as well as the functions that pertained to that particular class. After the classes where defined, I was able to go through the code and simplify the functions and code.
- Difficulties
    - It would have been cleaner code had I started with classes rather than trying to convert what I already had.
    - The value points of the Ace gave me the most problems. When and how do you count it as 1 or 11 in code.

3. Conclusions

- Somethings I have learned during this project.

    - Designing on paper before writing code, keeps the code tighter and less rewriting code.
    - I have clearer understanding of the data structures and how they are used.
- Shortcomings of the project.
    - Display the graphics for the card suits rather than displaying the word is a good feature.
    - The screen layout and reprinting everything every time is not efficient and looks bad.
- Features of the project
    - Quitting the program the correct way.
    - Cleaner code over the function approach and shorter. Easier to read and follow.
- Things I would do differently.
    - It would have been easier to start with classes and writing the code from that point instead of trying to convert my function based code to use classes.
    - I would have left the game play as a function rather than converting it to a class.
- Future options and features:
    - Make it multi player
    - Use additional decks
    - Add betting and wagering
    - Add a setup to configure the above options