CptS 121 - Program Design and Development

Programming Assignment 7: Poker (5-Card Draw)

Assigned: Wednesday, April 4, 2018
Due: Friday, April 13, 2018 by midnight

I. Learner Objectives:

At the conclusion of this programming assignment, participants should be able to:

- Declare and define arrays of pointers to strings
- Manipulate 2-Dimensional arrays
- Apply triply nested loops
- Manipulate strings

II. Prerequisites:

Before starting this programming assignment, participants should be able to:

- Apply and implement pointers in C
- Pass output parameters to functions
- Analyze a basic set of requirements and apply top-down design principles for a problem
- Apply repetition structures within an algorithm
- Construct while (), for (), or do-while () loops in C
- Compose C programs consisting of sequential, conditional, and iterative statements
- Eliminate redundancy within a program by applying loops and functions
- Create structure charts for a given problem
- Open and close files
- Read, write to, and update files
- Manipulate file handles
- Apply standard library functions: fopen (), fclose (), fscanf (), and fprintf ()
- Apply and implement pointers 2-dimenional arrays
- Define and apply structs in C
- Compose decision statements ("if" conditional statements)
- Create and utilize compound conditions
- Summarize topics from Hanly & Koffman Chapter 8 including:
  - What is an array?
  - Distinguishing between single dimensional and 2-dimentional arrays
  - What is an index?

III. Overview & Requirements:

Write a program that allows a user to play 5-Card-Draw Poker against the computer.

Start with the following example code supplied by Deitel & Deitel (example code). This will help you get started with the game of Poker. Please read this site to learn the rules of Poker http://en.wikipedia.org/wiki/5_card_draw. Complete the following step and you will have a working Poker game!!!

Adapted from Deitel & Deitel's C How to Program (6th Edition):

(1) In order to complete the game of 5-card-draw poker, you should complete the following functions:
(a) (5 pts) Modify the card dealing function provided in the example code so that a five-card poker hand is dealt.
(b) (5 pts) Write a function to determine if the hand contains a pair.
(c) (5 pts) Write a function to determine if the hand contains two pairs.
(d) (5 pts) Write a function to determine if the hand contains three of a kind (e.g. three jacks).
(e) (5 pts) Write a function to determine if the hand contains four of a kind (e.g. four aces).
(f) (5 pts) Write a function to determine if the hand contains a flush (i.e. all five cards of the same suit).
(g) (5 pts) Write a function to determine if the hand contains a straight (i.e. five cards of consecutive face values).

(2) (20 pts) Use the functions developed in (1) to deal two five-card poker hands, evaluate each hand, and determine which is the better hand.

(3) (25 pts) Simulate the dealer. The dealer's five-card hand is dealt "face down" so the player cannot see it. The program should then evaluate the dealer's hand, and based on the quality of the hand, the dealer should draw one, two, or three more cards to replace the corresponding number of unneeded cards in the original hand. The program should then re-evaluate the dealer's hand.

(4) (10 pts) Make the program handle the dealer's five-card hand automatically. The player should be allowed to decide which cards of the player's hand to replace. The program should then evaluate both hands and determine who wins. Now use the program to play 10 games against the computer. You should be able to test and modify or refine your Poker game based on these results!

You may make any adjustments or customizations to your Poker game that you wish!!! Have fun with this assignment!!!

IV. Submitting Assignments:

1. Using the OSBLE+ MS VS plugin, please submit your solution. Please visit https://github.com/WSU-HELPLAB/OSBLE/wiki/Submitting-an-Assignment for more information about submitting using OSBLE+.
2. Your project must contain one header file (a .h file), two C source files (which must be .c files), and project workspace.
3. Your project must build properly. The most points an assignment can receive if it does not build properly is 65 out of 100.

V. Grading Guidelines:

This assignment is worth 100 points. Your assignment will be evaluated based on a successful compilation and adherence to the program requirements. We will grade according to the following criteria:

1. (90 pts) for adherence to the above requirements (please see the individual point totals above)
2. (10 pts) for "good" style and design (i.e. proper function declarations, definitions, and comments)