# 

**Project 1**

**<’War’ Card Game>**

**CSC5 - 48102**

**Name: Cardenas, Diana “Charlie”**

**Date: 10/23/2016**

# 

# Introduction

Title: ‘War’ Card Game

‘War’ is a card game where the deck is divided evenly to the two players, giving each person a stack of cards. In unison, each player reveals the top card of their deck and enters into “battle,” where the player with the higher card takes both of the cards played and moves them to their stack. Aces are high, and suits are ignored.

When both players have the same card, the two go into “war.” In war, the two players draw another 3 card, the first two face down and the third one, face up. Whoever has the higher card at the last draw wins all the cards of that “war.”

The player with the most amount of cards at the end, wins.

## Summary

For this program, I tried to figure out a way to incorporate function calling. Instead of printing out just the card value, I decided to go a little more further and I took time to actually produce the image of the card when the computer or player would withdraw it. A month later into the class, I know I can shave off 100-200 lines of code off of this program.

The time it took to create this program was excruciating but the satisfaction of running the program successfully every now and then kept me going.

The writing of this program helped me to understand the random number generator and the importance of having else if, for loops, case, and do while loops in order to produce many different outcomes.

**Description**

The main point of this program is to utilize switch case to display the outcome of the rand() for each card drawn. It was also important to have a for loop and a else if loop in order for the player to play as long/short as they wanted to.

**Flowchart**

See Attachment

**Pseudo Code**

*Initialize*

*Prompt User to Exit or Stay*

*Case user enters “R”*

*Call the gameRule() function;*

*Return to menu*

*Else if user enters “Exit”*

*Exit the program*

*Run exit(0);*

*Case if user enters “p”*

*Call the gamePlay() function;*

*If gamePly is equal to “D”*

*Enter Number of rounds*

*For a equals 1, a less than and equal to rounds, add 1 to a after every round*

*If play is equal to “P”*

*Input random card value for pCard and cCard*

*If cCard is greater than pCard*

*cWin=1*

*else if pCard is greater then cCard*

*pWin=1*

*else (pCard is equal to cCard)*

*do*

*if war is equal to “f”*

*Input random card value for pCard and cCard*

*If cCard is greater than pCard*

*cWin=4*

*else if pCard is greater then cCard*

*pWin=4*

*else (war equals to “L”)*

*cWin=1*

*cCard=2*

*pCard=1*

*while cCard equals to pCard*

*else (play is equal to “q”)*

*If cPoints is greater than pPoints*

*Print computer as the winner*

*Else if pPoints is greater than cPoints*

*Print player as the winner*

*Else (tie)*

*Print the game was a draw*

*Leave equals to anything on the kyboard*

*Exit Program via exit(0)*

*Add up final points after each round*

*cPoints equals to cPoints plus cWin*

*pPoints equals to pPoints plus pWin*

*Else (Player inputs “Q”)*

*Exit program via exit(0)*

*If cPoints is greater than pPoints*

*Print computer as the winner*

*Else if pPoints is greater than cPoints*

*Print player as the winner*

*Else (tie)*

*Print the game was a draw*

*Return to the Menu*

**Major Variables**

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Variable Name** | **Description** | **Location** |
| Character | menuItm | The menu choice the user inputs | Int main |
|  | gamePly | Cin, asks user to quit the program via exit(0) or deal the deck | playGame() |
|  | play | The cin that asks user to play the round or to quit that game, exiting program after showing results |  |
|  | leave | The cin after showing results that prompts user to input anything in order to exit(0) |  |
|  | war | During a “war”, player choice to accept defeat (or continue with war |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |