Project 1 Write Up

<Blackjack>

CIS-5 46732

Name: Alan Cano

Date: 7/20/2020

**Version 1.0.0**

-Card Shuffling Included

**Version 2.0.0**

-Included Rules

-Removed other cards

-Set enemy and player

-Set deck faces to integers

-Added Win, Loss, and Tie

**Version 3.0.0**

-Improved card face designation

-Improved Win, Loss, Tie function

-Added Score

**Version 4.0.0**

-Added Points

-Added average system

**Introduction**

Title: Blackjack

This is a card game based off of blackjack.

The user clicks Y to start is given a randomized pair of cards. If they feel good, they can go against the enemy player to see which has the bigger value.

Ex: Player: Ace + Ace = 2 < Enemy: Two + Two = 4

This game is game of chance against a randomized program.

**Summary**

Project Size: About 170 lines

Number of variables: 25

This project utilizes the concepts learned in lectures and labs especially emphasizing on do-while loops, switch cases, bools, and if-else statements.

**Description**

The player is given an option to read the rules or immediately engage in the game. Once the rules are read, the game starts and the user is given a random pair of cards.

**Pseudo Code**

*Initialize*

*If rules is selected*

*Output Rules text*

*Else*

*Start Game*

*Shuffle cards*

*Assign titles to player card 1*

*Assign titles to player card 2*

*Display player cards*

*If satisfied with cards*

*Shuffle enemy cards*

*Assign titles to enemy card 1*

*Assign titles to enemy card 2*

*Display enemy cards*

*Else*

*Shuffle and display cards again until ‘Y’ is selected*

*Calculate sum of values of cards*

*Calculate larger sum*

*If players chooses again*

*Game restarts*

*Else*

*Game ends*

*Asks for three scores and number of games*

*Displays average and ends program*

**Program**

#include <iostream> //I/O Library

#include <cstdlib> //Random Function Library

#include <ctime> //Time Library

#include <iomanip> //Formatting Library

#include <fstream> //File input/output

#include <cmath> //Math Library

using namespace std;

/\*

\*

\*/

int main(int argc, char\*\* argv) {

//File input/output

srand(static\_cast<unsigned int>(time(0)));

//Declare Variable Data Types and Constants

int score, points;

string file;

float a,b,c,pts,gam,average,avg;

bool playWin;

bool enemWin;

unsigned short vC1,vC2, vCe1, vCe2;

unsigned char nCards,nEcards, PlaySum, EnemSum;

char option;

char chance, inp1, inp2;

ifstream fileC;

//Initialize Variables

score = 0;

nCards=13;

nEcards=13; //number of enemy cards

do{

cout <<"Welcome to BlackJack!" << endl;

cout<<"Would you like to read the rules? Y/N"<<endl;

cin>>option;

if ((option=='Y')||(option=='y')){ //validating user input

fileC.open("Project 1 Rules V4.txt",ios::in); //open a file to

//perform read operation using file object

if (fileC.is\_open()){ //checking whether the file is open

string tp;

while(getline(fileC, tp)){ //read data from file object and put it into string.

cout << tp << "\n"; //print the data of the string

}

fileC.close(); //close the file object.

}

}

//validating user input

do{

bool playWin = false;//initialize both bools to false

bool enemWin = false;//only switch to true if they get win condition

cout << "Player cards are ";

vC1=rand()%nCards+1;

//Unique Value for first 2 cards

do{

vC2=rand()%nCards+1;

}while(vC1==vC2);

//Order the first 2

if(vC1>vC2){

unsigned short temp=vC1;

vC1=vC2;

vC2=temp;

}

vCe1=rand()%nCards+1;

do{

vCe2=rand()%nCards+1;

}while(vCe1==vCe2);

//Order the first 2

if(vCe1>vCe2){

unsigned short temp2=vCe1;

vCe1=vCe2;

vCe2=temp2;

}

if (vC1==1)cout<<"Ace ";

if (vC1==2)cout<<"Two ";

if (vC1==3)cout<<"Three ";

if (vC1==4)cout<<"Four ";

if (vC1==5)cout<<"Five ";

if (vC1==6)cout<<"Six ";

if (vC1==7)cout<<"Seven ";

if (vC1==8)cout<<"Eight ";

if (vC1==9)cout<<"Nine ";

if (vC1==10)cout<<"Ten ";

if (vC1==11)cout<<"Jack ";

if (vC1==12)cout<<"Queen ";

if (vC1==13)cout<<"King ";

if (vC2==1)cout<<"and Ace";

if (vC2==2)cout<<"and Two";

if (vC2==3)cout<<"and Three";

if (vC2==4)cout<<"and Four";

if (vC2==5)cout<<"and Five";

if (vC2==6)cout<<"and Six";

if (vC2==7)cout<<"and Seven";

if (vC2==8)cout<<"and Eight";

if (vC2==9)cout<<"and Nine";

if (vC2==10)cout<<"and Ten";

if (vC2==11)cout<<"and Jack";

if (vC2==12)cout<<"and Queen";

if (vC2==13)cout<<"and King";

cout<<"\nPlay your cards?";

cin>>option;

}while ((option=='N')||(option=='n'));

if ((option=='Y')||(option=='y')){

cout<< "Your enemy cards are: ";

if(vCe1==13)cout<<"King";

else if(vCe1==12)cout<<"Queen";

else if(vCe1==11)cout<<"Jack";

else if(vCe1==10)cout<<"Ten";

else if(vCe1==9)cout<<"Nine";

else if(vCe1==8)cout<<"Eight";

else if(vCe1==7)cout<<"Seven";

else if(vCe1==6)cout<<"Six";

else if(vCe1==5)cout<<"Five";

else if(vCe1==4)cout<<"Four";

else if(vCe1==3)cout<<"Three";

else if(vCe1==2)cout<<"Two";

else if(vCe1==1)cout<<"One";

switch(vCe2){

case 1: cout<<" and Ace";break;

case 2: cout<<" and Two";break;

case 3: cout<<" and Three";break;

case 4: cout<<" and Four";break;

case 5: cout<<" and Five";break;

case 6: cout<<" and Six";break;

case 7: cout<<" and Seven";break;

case 8: cout<<" and Eight";break;

case 9: cout<<" and Nine";break;

case 10: cout<<" and Ten";break;

case 11: cout<<" and Jack";break;

case 12: cout<<" and Queen";break;

case 13: cout<<" and King";break;

}

};

PlaySum = vC1+vC2;

EnemSum = vCe1 + vCe2;

PlaySum <= EnemSum?enemWin=true:enemWin=false;

PlaySum >= EnemSum?playWin=true:playWin=false;

if(!enemWin){

score++;

points = score\*10;

cout<<"\nPlayer Wins!"<<endl;

cout << "Congrats! You have won "<< score << " time(s)" <<

"and earned " << points << " points!" << endl;

}

else if(!playWin)cout<<"\nEnemy Wins!"<<endl;

else cout<<"It is a Tie!" <<endl;

cout<<"\nAgain? Y/N"<<endl; //nested loop

cin>>option;

if ((option=='Y')||(option=='y'))

cout << endl;

}while((option=='Y')||(option=='y'));

cout << "Input three scores you earned "

"and how many games you played"<<endl;

cin>>a>>b>>c>>gam;

average =(a+b+c)/gam;

avg = pow(average, 2.0);

cout << "Average success is "<< setw(8) << avg << "%";

//Display Outputs

return 0;

}