Project 1

Blackjack

CSC-5-46332
Brandon Smith
07/25/2022

Introduction

Title: Blackjack
This is a card game.
The goal of the game is to get as close to 21 as possible without going over. If you go over, you bust and automatically lose. You play against the dealer, they must draw if they are under 17.
The controls are simple, type ‘H’ if you want to hit and draw a card or ‘S’ if you want to stand.

Summary

The project is approximately 260 lines.
There are 26 variables in this version.

This project uses most of the techniques learned throughout this course. This project all has a lot of potential for project 2. The advanced techniques to be utilized in project 2 will allow the code to be much more efficient as well as fully playable.

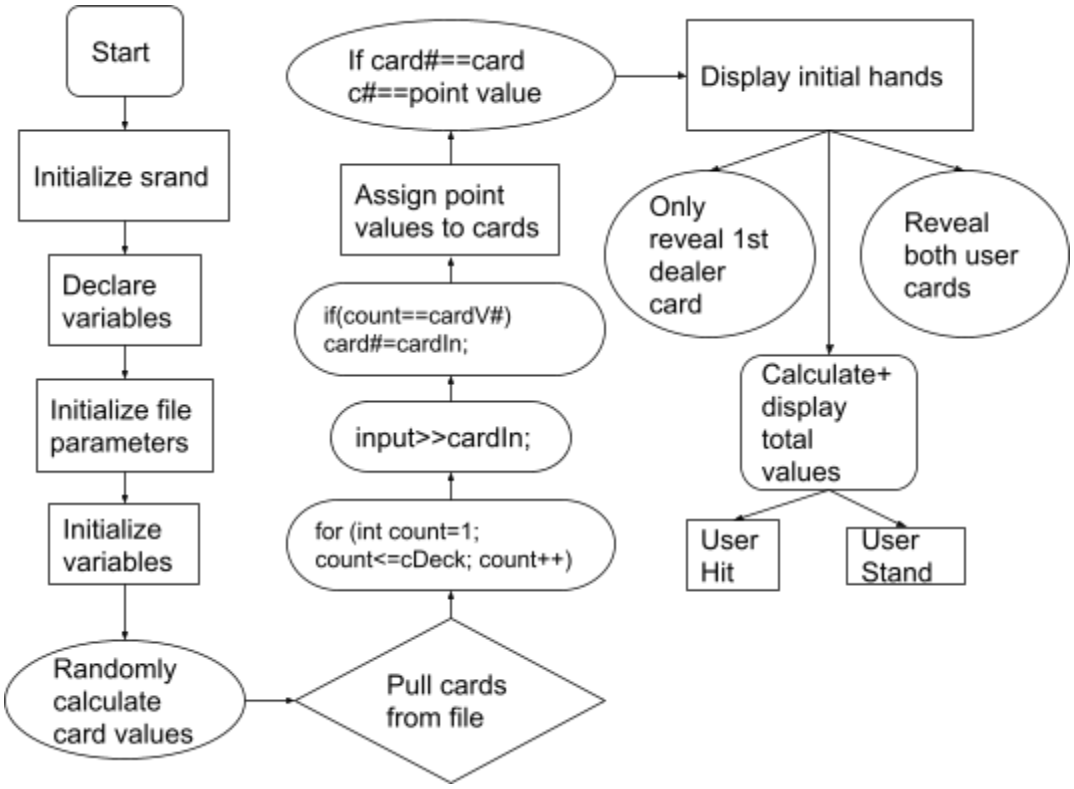
This project took a lot of work to get functioning, for example the process of drawing cards took me nearly a full day. There are plenty of bugs and features yet to be included and as a result I am not proud of this project. I left the bugs fixing for project 2 since most of the code will be rewritten using the newer techniques. I will make sure project 2 is something I am proud to present with all the desired features and (little to) no bugs.

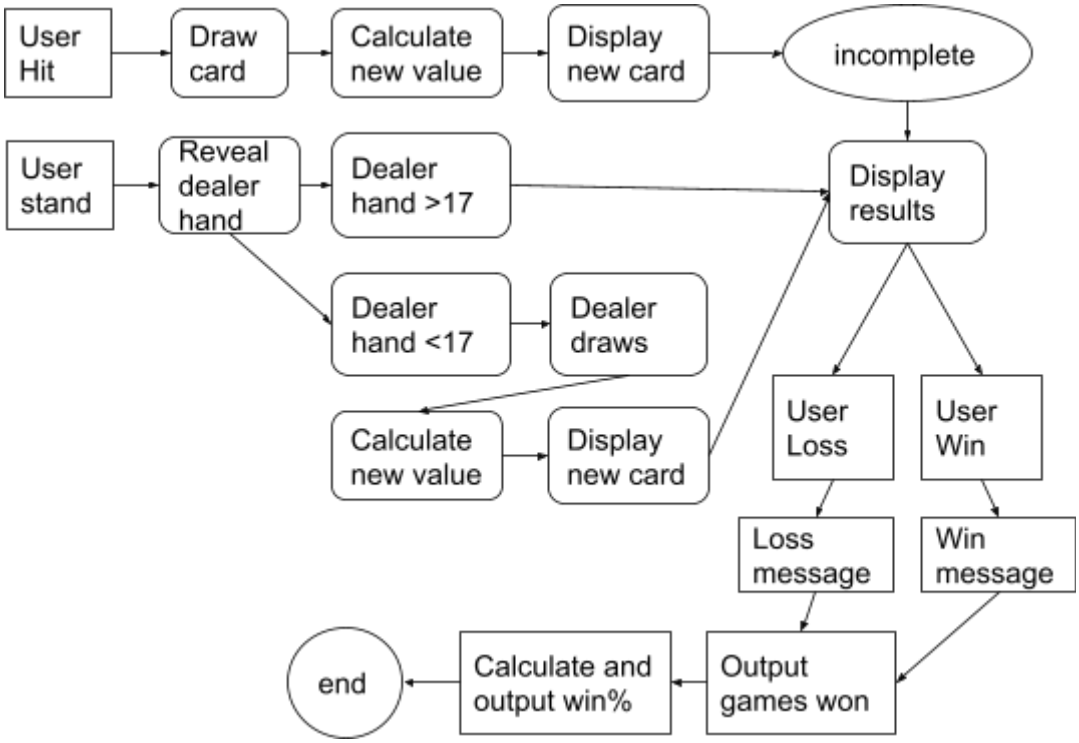
The next project will use arrays and functions to streamline most processes.

Description

The main point of this program is to respond to the user’s input and calculate the score of the user and dealer and output a winner.

Flow Chart


Start
Initialize srand
Declare variables
Initialize file parameters
Initialize variables
Randomly calculate card values
Pull cards from file
for (int count=1; count<=cDeck; count++)
input>>cardIn;
if(count==cardV#) card#=cardIn;
Assign point values to cards
If card#==card c#==point value
Display initial hands
Only reveal 1st dealer card
Reveal both user cards
Calculate+ display total values
User Hit
User Stand


User Hit
Draw card
Calculate new value
Display new card
incomplete
User stand
Reveal dealer hand
Dealer hand >17
Display results
Dealer hand <17
Dealer draws
Calculate new value
Display new card
User Loss
User Win
Loss message
Win message
end
Calculate and output win%
Output games won

Pseudo Code

Initialize scrad
Declare variables
Initialize file parameters
Initialize variables

Randomly calculate card values
Pull Cards from file
    For loop
    If count is card value, assign card

Calculate total value of hand
If statement for each point value of cards
Add first two user cards to user total
Add first dealer card to dealer total

Display initial hand
Show dealer's first card and value
Show user's first two cards and total value
Input user hit or stand

Switch
Case Hit
Deal extra card for user
Incomplete

Case Stand
Reveal dealer's hand and total value
If dealer total<17, dealer draws card
If dealer total>17, dealer stands

If dealer total>21, dealer bust, user wins
If dealer total<21 and less than user total, user wins
If dealer total<21 and more than user total, user loses

Output results
Output win/loss message
Display how many games won
Calculate win/loss percentage
Display win/loss percentage

Major Variables

Type	Variable Name	Description	Location (line#)
fstream	input	File input	29
string	filename	Name of file	30
	card1-3, dcard1-3	Cards of user and dealer	31-32
	cardIn	Card input from file	66
unsigned short	cardV1-3, dcardV1-3	Value of user and dealer cards	33-34
int	cDeck	Cards in deck	35
	userT, dealT	Total of user and dealer cards	36
	gWon	Games won	37
	gPlay	Games played	37
	winP	Win percentage	38
	c1-3, d1-3	Point values of user and dealer cards	82
char	hOs	Hit or Stand input	39
bool	uBust, dBust	User or dealer bust	40
	win	Did user win	40

Textbook Concepts

Chapter	Keywords	Location (line#)
2	libraries	10-16
	variables/literals	29-40, 66, 82
	integers	35-38, 82
	characters	39
	string	30, 66, 241
	floats	249
	bools	40, 229, 234, 243-245
	comments	N/A
3	Type casting	249
	Formatting output	176, 178, 194, 196, 207, 209, 223, 225, 249
	strings	242-245
4	if	70-75, 209, 225, 230
	if-else	201, 216
	if-else-if	85-171
	nesting	69-75, 209, 225, 230
	logical operators	85-171
	validating user input	182
	conditional operator	67
	switch	187
5	increment/decrement	212, 228, 233
	while	182
	For loop	67
	File input	42-43

Program

```
/*
 * File:   main.cpp
 * Author: Brandon Smith
 * Created on July 22
 * Purpose: Blackjack v2, now displays results of game.
 *          User can only stand for "proper" results
 */

//System Libraries
#include <iostream>     //Input/Output library
#include <iomanip>      //Format Library
#include <cmath>        //Math Library
#include <cstdlib>      //Srand
#include <fstream>      //File operator
#include <string>       //String Library
#include <ctime>        //Time to set random number seed
using namespace std;

//User Libraries
//Global Constants
//Mathematical/Physics/Conversions, Higher dimensioned arrays
//Function Prototypes

//Execution Begins Here
int main(int argc, char** argv) {
    //Initialize the Random Number Seed
    srand(static_cast<unsigned>(time(0)));
    //Declare Variables
    fstream input;                      //File input
    string fileName,                    //File name
           card1, card2, card3,         //Cards 1-3 String
           dCard1, dCard2, dCard3;      //Dealer Cards 1-3 String
    unsigned short cardV1, cardV2, cardV3,   //Cards 1-3
                   dCardV1, dCardV2, dCardV3;//Dealer Cards 1-3
    int cDeck,                          //Cards in deck
        userT, dealT,                   //User total, dealer total
        gWon, gPlay,                    //Games won, game played
        winP;                           //Win Percentage
    char hOs;                           //hit or stand
```

```
bool uBust, dBust, win;                //User bust, dealer bust, game result
//Initialize file parameters
fileName="deckOfCards.dat";            //File name
input.open(fileName.c_str(),ios::in);  //Opens file
//Initialize Variables
cDeck=52;
userT=0;                //Default user total
dealT=0;                //Default dealer total
gWon=0;                 //Default games won
winP=0;                 //Default win percentage
gPlay=0;                //Default games played
uBust=false;            //Default user bust
dBust=false;            //Default dealer bust
win=false;              //Default game result
/*
 *Technically, 21 cards can be drawn but the code
 * would be too bloated without arrays so the limit is 1 extra cards
 */
cardV1=(rand()%cDeck+1);
cardV2=(rand()%cDeck+1);
cardV3=(rand()%cDeck+1);
dCardV1=(rand()%cDeck+1);
dCardV2=(rand()%cDeck+1);
dCardV3=(rand()%cDeck+1);

//Pull cards from the file
string cardIn;
for (int count=1; count<=cDeck; count++)
{
   input>>cardIn;
   if (count==cardV1)card1=cardIn;
   if (count==cardV2)card2=cardIn;
   if (count==cardV3)card3=cardIn;
   if (count==dCardV1)dCard1=cardIn;
   if (count==dCardV2)dCard2=cardIn;
   if (count==dCardV3)dCard3=cardIn;
}

/*
 * Calculate total value of hand
```

```
 * Arrays will make this much more efficient in project 2
*/
int c1, c2, c3, dc1, dc2, dc3;      //Point values of cards 1-3 User and Dealer


if (cardV1==1  or cardV1==14  or cardV1==27  or cardV1==40)c1=11;
else if (cardV1==2  or cardV1==15  or cardV1==28  or cardV1==41)c1=2;
else if (cardV1==3  or cardV1==16  or cardV1==29  or cardV1==42)c1=3;
else if (cardV1==4  or cardV1==17  or cardV1==30  or cardV1==43)c1=4;
else if (cardV1==5  or cardV1==18  or cardV1==31  or cardV1==44)c1=5;
else if (cardV1==6  or cardV1==19  or cardV1==32  or cardV1==45)c1=6;
else if (cardV1==7  or cardV1==20  or cardV1==33  or cardV1==46)c1=7;
else if (cardV1==8  or cardV1==21  or cardV1==34  or cardV1==47)c1=8;
else if (cardV1==9  or cardV1==22  or cardV1==35  or cardV1==48)c1=9;
else if (cardV1==10  or cardV1==23  or cardV1==36  or cardV1==49)c1=10;
else if (cardV1==11  or cardV1==24  or cardV1==37  or cardV1==50)c1=10;
else if (cardV1==12  or cardV1==25  or cardV1==38  or cardV1==51)c1=10;
else if (cardV1==13  or cardV1==26  or cardV1==39  or cardV1==52)c1=10;
userT+=c1;

if (cardV2==1  or cardV2==14  or cardV2==27  or cardV2==40)c2=11;
else if (cardV2==2  or cardV2==15  or cardV2==28  or cardV2==41)c2=2;
else if (cardV2==3  or cardV2==16  or cardV2==29  or cardV2==42)c2=3;
else if (cardV2==4  or cardV2==17  or cardV2==30  or cardV2==43)c2=4;
else if (cardV2==5  or cardV2==18  or cardV2==31  or cardV2==44)c2=5;
else if (cardV2==6  or cardV2==19  or cardV2==32  or cardV2==45)c2=6;
else if (cardV2==7  or cardV2==20  or cardV2==33  or cardV2==46)c2=7;
else if (cardV2==8  or cardV2==21  or cardV2==34  or cardV2==47)c2=8;
else if (cardV2==9  or cardV2==22  or cardV2==35  or cardV2==48)c2=9;
else if (cardV2==10  or cardV2==23  or cardV2==36  or cardV2==49)c2=10;
else if (cardV2==11  or cardV2==24  or cardV2==37  or cardV2==50)c2=10;
else if (cardV2==12  or cardV2==25  or cardV2==38  or cardV2==51)c2=10;
else if (cardV2==13  or cardV2==26  or cardV2==39  or cardV2==52)c2=10;
userT+=c2;

if (cardV3==1  or cardV3==14  or cardV3==27  or cardV3==40)c3=11;
else if (cardV3==2  or cardV3==15  or cardV3==28  or cardV3==41)c3=2;
else if (cardV3==3  or cardV3==16  or cardV3==29  or cardV3==42)c3=3;
else if (cardV3==4  or cardV3==17  or cardV3==30  or cardV3==43)c3=4;
else if (cardV3==5  or cardV3==18  or cardV3==31  or cardV3==44)c3=5;
```

```
else if (cardV3==6  or cardV3==19  or cardV3==32  or cardV3==45)c3=6;
else if (cardV3==7  or cardV3==20  or cardV3==33  or cardV3==46)c3=7;
else if (cardV3==8  or cardV3==21  or cardV3==34  or cardV3==47)c3=8;
else if (cardV3==9  or cardV3==22  or cardV3==35  or cardV3==48)c3=9;
else if (cardV3==10  or cardV3==23  or cardV3==36  or cardV3==49)c3=10;
else if (cardV3==11  or cardV3==24  or cardV3==37  or cardV3==50)c3=10;
else if (cardV3==12  or cardV3==25  or cardV3==38  or cardV3==51)c3=10;
else if (cardV3==13  or cardV3==26  or cardV3==39  or cardV3==52)c3=10;

if (dCardV1==1  or dCardV1==14  or dCardV1==27  or dCardV1==40)dc1=11;
else if (dCardV1==2  or dCardV1==15  or dCardV1==28  or dCardV1==41)dc1=2;
else if (dCardV1==3  or dCardV1==16  or dCardV1==29  or dCardV1==42)dc1=3;
else if (dCardV1==4  or dCardV1==17  or dCardV1==30  or dCardV1==43)dc1=4;
else if (dCardV1==5  or dCardV1==18  or dCardV1==31  or dCardV1==44)dc1+=5;
else if (dCardV1==6  or dCardV1==19  or dCardV1==32  or dCardV1==45)dc1=6;
else if (dCardV1==7  or dCardV1==20  or dCardV1==33  or dCardV1==46)dc1=7;
else if (dCardV1==8  or dCardV1==21  or dCardV1==34  or dCardV1==47)dc1=8;
else if (dCardV1==9  or dCardV1==22  or dCardV1==35  or dCardV1==48)dc1=9;
else if (dCardV1==10  or dCardV1==23  or dCardV1==36  or dCardV1==49)dc1=10;
else if (dCardV1==11  or dCardV1==24  or dCardV1==37  or dCardV1==50)dc1=10;
else if (dCardV1==12  or dCardV1==25  or dCardV1==38  or dCardV1==51)dc1=10;
else if (dCardV1==13  or dCardV1==26  or dCardV1==39  or dCardV1==52)dc1=10;
dealT+=dc1;

if (dCardV2==1  or dCardV2==14  or dCardV2==27  or dCardV2==40)dealT+=11;
else if (dCardV2==2  or dCardV2==15  or dCardV2==28  or dCardV2==41)dc2=2;
else if (dCardV2==3  or dCardV2==16  or dCardV2==29  or dCardV2==42)dc2=3;
else if (dCardV2==4  or dCardV2==17  or dCardV2==30  or dCardV2==43)dc2=4;
else if (dCardV2==5  or dCardV2==18  or dCardV2==31  or dCardV2==44)dc2=5;
else if (dCardV2==6  or dCardV2==19  or dCardV2==32  or dCardV2==45)dc2=6;
else if (dCardV2==7  or dCardV2==20  or dCardV2==33  or dCardV2==46)dc2=7;
else if (dCardV2==8  or dCardV2==21  or dCardV2==34  or dCardV2==47)dc2=8;
else if (dCardV2==9  or dCardV2==22  or dCardV2==35  or dCardV2==48)dc2=9;
else if (dCardV2==10  or dCardV2==23  or dCardV2==36  or dCardV2==49)dc2=10;
else if (dCardV2==11  or dCardV2==24  or dCardV2==37  or dCardV2==50)dc2=10;
else if (dCardV2==12  or dCardV2==25  or dCardV2==38  or dCardV2==51)dc2=10;
else if (dCardV2==13  or dCardV2==26  or dCardV2==39  or dCardV2==52)dc2=10;
dealT+=dc2;

if (dCardV3==1  or dCardV3==14  or dCardV3==27  or dCardV3==40)dc3=11;
```

```
else if (dCardV3==2  or dCardV3==15  or dCardV3==28  or dCardV3==41)dc3=2;
else if (dCardV3==3  or dCardV3==16  or dCardV3==29  or dCardV3==42)dc3=3;
else if (dCardV3==4  or dCardV3==17  or dCardV3==30  or dCardV3==43)dc3=4;
else if (dCardV3==5  or dCardV3==18  or dCardV3==31  or dCardV3==44)dc3+=5;
else if (dCardV3==6  or dCardV3==19  or dCardV3==32  or dCardV3==45)dc3=6;
else if (dCardV3==7  or dCardV3==20  or dCardV3==33  or dCardV3==46)dc3=7;
else if (dCardV3==8  or dCardV3==21  or dCardV3==34  or dCardV3==47)dc3=8;
else if (dCardV3==9  or dCardV3==22  or dCardV3==35  or dCardV3==48)dc3=9;
else if (dCardV3==10  or dCardV3==23  or dCardV3==36  or dCardV3==49)dc3=10;
else if (dCardV3==11  or dCardV3==24  or dCardV3==37  or dCardV3==50)dc3=10;
else if (dCardV3==12  or dCardV3==25  or dCardV3==38  or dCardV3==51)dc3=10;
else if (dCardV3==13  or dCardV3==26  or dCardV3==39  or dCardV3==52)dc3=10;

//Display the initial Hands
cout<<"-----------------------------------------------"<<endl;
cout<<"Dealer's Hand: "<<dc1<<" + ?"<<endl;
cout<<left<<setw(12)<<dCard1<<left<<setw(12)<<" ? "<<endl;
cout<<"Your Hand: "<<userT<<endl;
cout<<left<<setw(12)<<card1<<left<<setw(12)<<card2<<endl;
cout<<"-----------------------------------------------"<<endl;
cout<<"Would you like to hit or stand?"<<endl;
cin>>hOs;
while (hOs!='H' and hOs!='h' and hOs!='S' and hOs!='s')
{
   cout<<"Invalid choice, please enter 'H' for hit or 'S' for stand."<<endl;
   cin>>hOs;
}
switch (hOs)
{
   case 'H':                  //Case Hit
   case 'h':
   userT+=c3;
   cout<<"-----------------------------------------------"<<endl;
   cout<<"Dealer's Hand: "<<dealT<<endl;
   cout<<left<<setw(12)<<dCard1<<left<<setw(12)<<dCard2<<endl;
   cout<<"Your Hand: "<<userT<<endl;
   cout<<left<<setw(12)<<card1<<left<<setw(12)<<card2<<setw(12)<<card3<<endl;
   cout<<"-----------------------------------------------"<<endl;

   case 'S':                  //Case Stand
```

```
        case 's':
        if (dealT>16)                   //If dealer is over 16
        {
           cout<<"-----------------------------------------------"<<endl;
           cout<<"Dealer's Hand: "<<dealT<<endl;
           cout<<left<<setw(12)<<dCard1<<left<<setw(12)<<dCard2<<endl;
           cout<<"Your Hand: "<<userT<<endl;
           cout<<left<<setw(12)<<card1<<left<<setw(12)<<card2<<endl;
           cout<<"-----------------------------------------------"<<endl;
           if (dealT<userT)
           {
             win=true;   //User wins if score is greater than dealer
             gWon++;
           }
        }

        else                            //If dealer is under 17, must draw
        {
           dealT+=dc3;
           cout<<"-----------------------------------------------"<<endl;
           cout<<"Dealer's Hand: "<<dealT<<endl;

cout<<left<<setw(12)<<dCard1<<left<<setw(12)<<dCard2<<left<<setw(12)<<dCard3<<
endl;
           cout<<"Your Hand: "<<userT<<endl;
           cout<<left<<setw(12)<<card1<<left<<setw(12)<<card2<<endl;
           cout<<"-----------------------------------------------"<<endl;
           if (dealT>21)
           {
              dBust=true;    //Dealer busts, user wins
              gWon++;
           }
           if (dealT<userT)
           {
              win=true;   //User wins if score is greater than dealer
              gWon++;
           }
        }
```

```
        //Results of Game
        string winH, winB, loss, draw;
        winH="You win, congratulations!";
        winB="Dealer busts! You win!";
        loss="You lost, better luck next time";
        draw="It's a draw!";
        gPlay++;
        if (dealT==userT)cout<<draw<<endl;                      //If draw
        else if (dBust==true)cout<<winB<<endl;                  //If dealer busts
        else if (win==true)cout<<winH<<endl;                    //If user won
        else if (win==false && dealT!=userT)cout<<loss<<endl;   //If user lost
        cout<<"You won "<<gWon<<" games out of "<<gPlay<<endl;
        winP=(gWon/gPlay)*100.00;                               //Calculate win%
        cout<<"Your win percentage is "
            <<fixed<<setprecision(2)<<static_cast<float>(winP)<<"%"<<endl;
    }
    //Exit stage right
    input.close();
        return 0;
}

/*
 * FUTURE CHANGES
 * Implement repeated plays (add on Project 2 with advanced techniques)
 * Implement hit for user, and repeated hits for dealer (Project 2)
 * Ace turns to 1 if over 21 (this is causing a few bugs currently)
 *
 * BUGS
 * Dealer hand value incorrect, usually large negative number
 * User will sometimes lose with larger value
 */
```