Project 2

Title

Blackjack

Course

CSC 5

Section

42450

Due Date

June 9, 2014

Author

Victor Medel

Introduction

Title: Blackjack

This version continues to be a simple program that allows any player to quickly play a game of Blackjack. The program starts up with a menu with three options, *1. Play Blackjack*, *2. Blackjack Game Overview,* or *Anything Else to Exit*. The options are very straight forward, option one allows the user to play the game, option two provide the quick overview of the game as written below, and any other input exits the user from program.

The object of the game is to beat the house by receiving a score of 21 or by getting a higher score than the house without going over 21 with any additional cards. The game begins by dealing two cards to the player; after displaying your score and if your score is less than 21 you will have the option to take another card to add to your total score or hold with your existing score. If you hold or go over 21 after choosing another card the program will automatically display the house's hand and then determine the outcome. Multiple decks of cards are used with the following values:

Cards 2 through 10 = face value points
Jacks = 10 points
Queens = 10 Points
Kings = 10 Points
Aces = 1 or 11 depends on the player's total. If player's total is less than ten points then they hold a value of eleven otherwise the card will hold a value of one.

Summary

Developing the program took over two weeks and several versions due my continued limited experience with C++ programing, speed of the class lectures, and just like Project 1, the project packet development. As references I used Project 1, all class lecture examples posted on GitHub, the course textbook (*Problem Solving with C++ by Walter Savitch*), the web to obtain some of the rules on how to play Blackjack as well as how to start this project. I also utilized the sample project documentation provided on Black Board to help me with the production of this document.

I've developed this program utilizing many of the concepts that have been covered by the class textbook (*Problem Solving with C++ 8th Edition by Walter Savitch*) within chapters one through seven. I have also used concepts discussed during class lecture and lab to create this program. The program runs as expected but I believe that this program still has many opportunities. The lack of time left in this semester has limited my ability to fully understand the final chapters that have been covered in this course. I would like to believe that I would continue to read the class textbook over the summer but the reality is that I will need to enroll in the next level of C++ programming classes to continue to learn, which I am planning on doing. My plan is to fully understand Arrays by the end of June and as far as the rest of the material, I will wait for CSC 17A.

One of the major obstacles that I encountered while developing this program was the ability to utilize a two dimensional array to deal a card; therefore, I developed the program to work with a one dimensional array only. I did include the code that I was attempting to work at the very end of my code, lines 361-392.

The course covered many basic concepts of C++ programing and I did try to include all of the concepts covered in class but due to the limited time to develop this program I mainly concentrated on the items listed on the assignment prompt which are the following:

a) Functions and Arrays [Lines 313-319]

b) One and Two Dimensional Arrays [Lines 30 & 31, 2D attempt on 364]
c) Passing Arrays Between Functions [Lines 313-326]
d) Pass by Value [Line 359]
e) Pass by Reference [Lines 313-326]
f) Defaulted Parameters [Lines 21-23]
g) Returning Primitive Data Types [Lines 359]
h) Formatting [Lines 323-325]
i) Reading and Writing to Files [Lines 59-67 & 69-78]

Concepts Used

From Textbook:
Problem Solving with C++ 8th Edition by Walter Savitch

Chapter 2
2.1 Variables and Assignments
2.2 Input and Output
2.3 Data Types and Expressions
2.4 Simple Flow Control
2.5 Program Style

Chapter 3
3.1 Using Boolean Expressions
3.2 Multiway Branches
3.3 More About C++ Loop Statements
3.4 Designing Loops

Chapter 4
4.1 Top-Down Design
4.2 Predefined Functions
4.3 Programmer-Defined Functions
4.4 Procedural Absraction
4.5 Scope And Local Variables

Chapter 5
5.1 void Functions
5.2 Call-By-Reference Parameters
~~5.3 Using Procedural Abstraction~~
5.4 Testing and Debugging Functions
5.5 General Debugging Techniques

Chapter 6
6.1 Streams and Basic File I/O
6.2 Tools for Stream I/O
6.3 Character I/O

Chapter 7
7.1 Introduction to Arrays
7.2 Arrays in Functions
7.3 Programming with Arrays
7.4 Multidimensional Array

From Class Lectures and Lab:

1. Input and Output
2. Loops
3. Menus
4. Branching Constructs
5. Mathematical Expressions
6. User interactivity
7. Boolean Expressions
8. Functions
9. One Dimensional Arrays
10. Two Dimensional Arrays
11. Void Functions
12. Programming Logic

Code Specifications

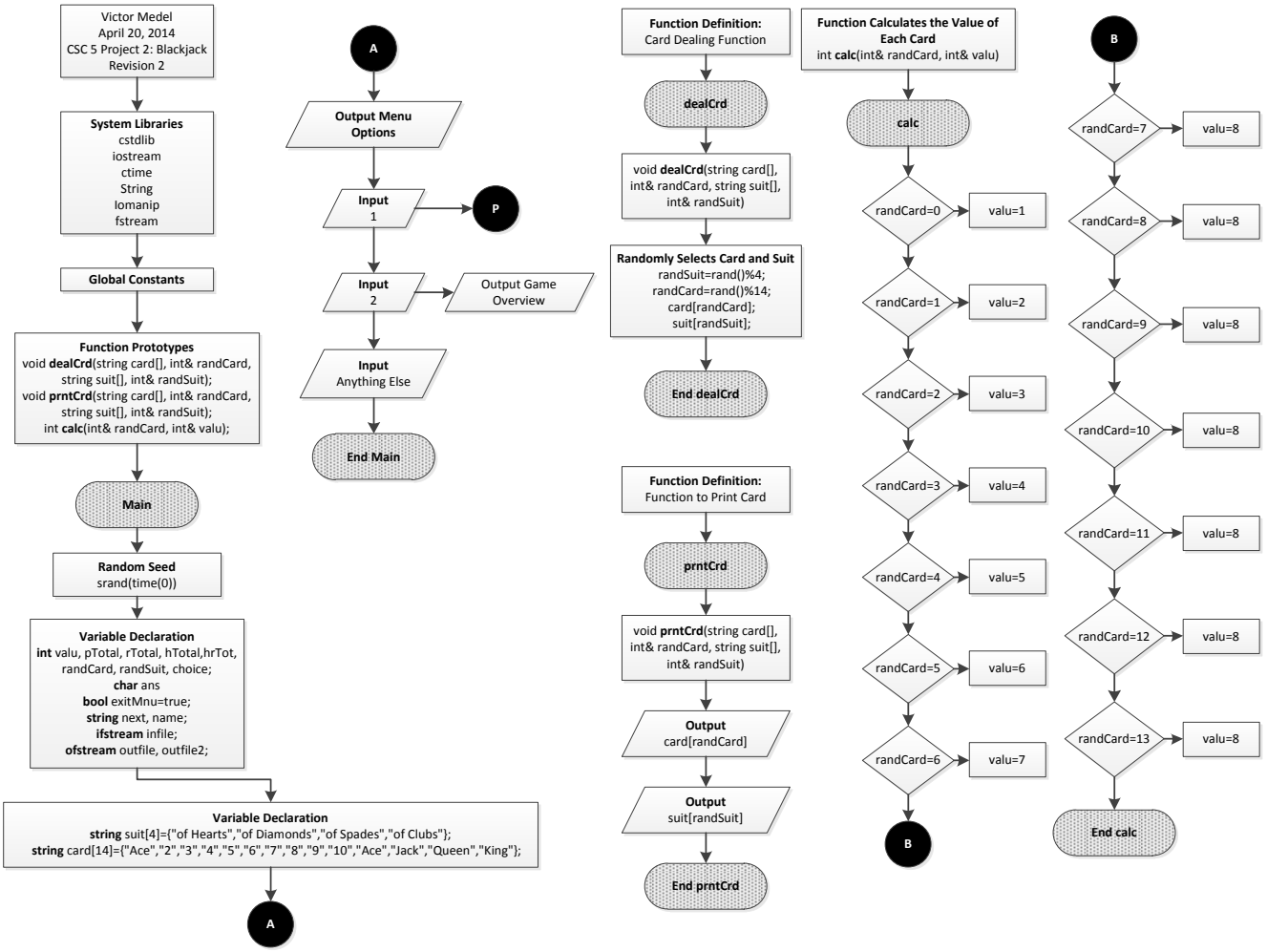
Lines of Code	**303**
Comment Lines	**84**
Blank Lines	**4**
Total Lines of Source Files	**391**
Number of Variables	**18**

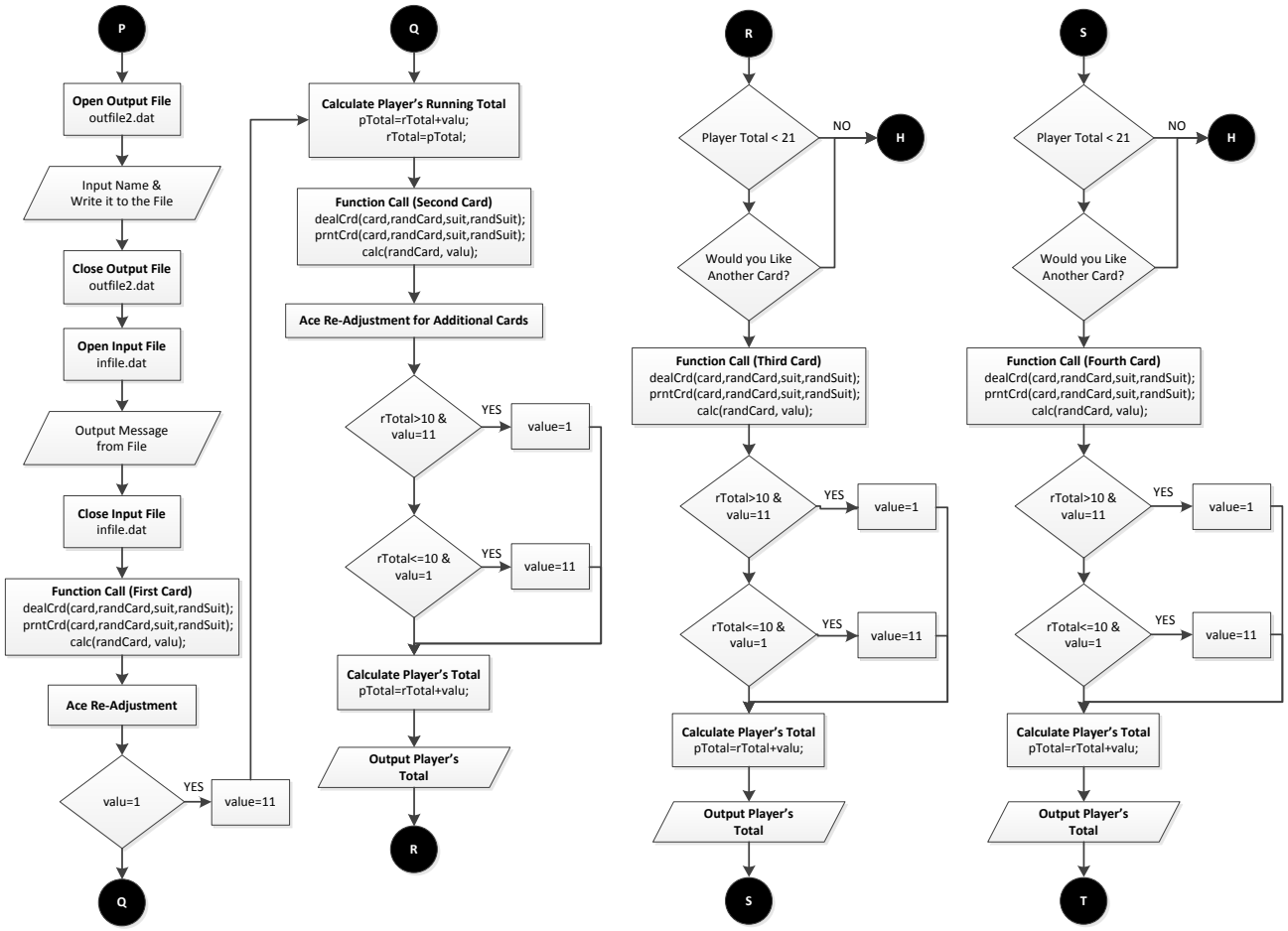
Variables Used

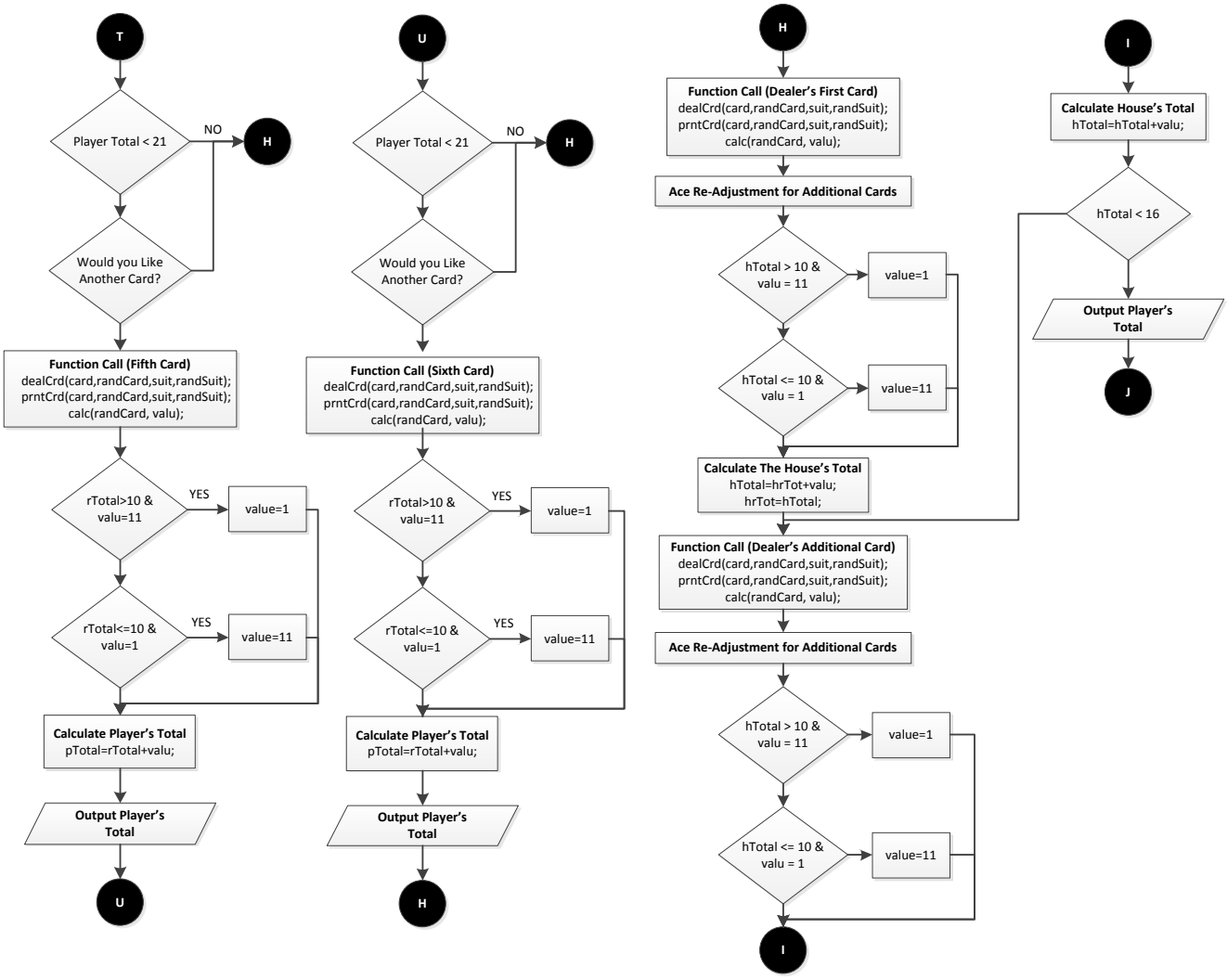
Type	Variable Name	Description
Integer	valu	Function parameter that hold the dealt card value within the main program
	pTotal	Holds the players total score within the main program. It is also used as a function parameter that holds the players total score within the function definition and function header
	rTotal	Utilized to keep a running total of the player score
	total	Function parameter used to hold a card value total
	hTot	Holds the house's total score
	hrTot	Utilized to keep a running total of the house's score
	choice	Menu selection input
	randCard	Randomly selects a number/card from 1 to 14
	randSuit	
		Randomly selects a number/suit from 1 to 4
String	card[]	Array that outputs a card value
	suit[]	Array that outputs a suit value
	next	Variable used to read from a file
	name	Variable used to write a name to a file
Ifstream	infile	Variable used to identify file that is being written to
Ofstream	outfile	Variable used to identify file that is being read from

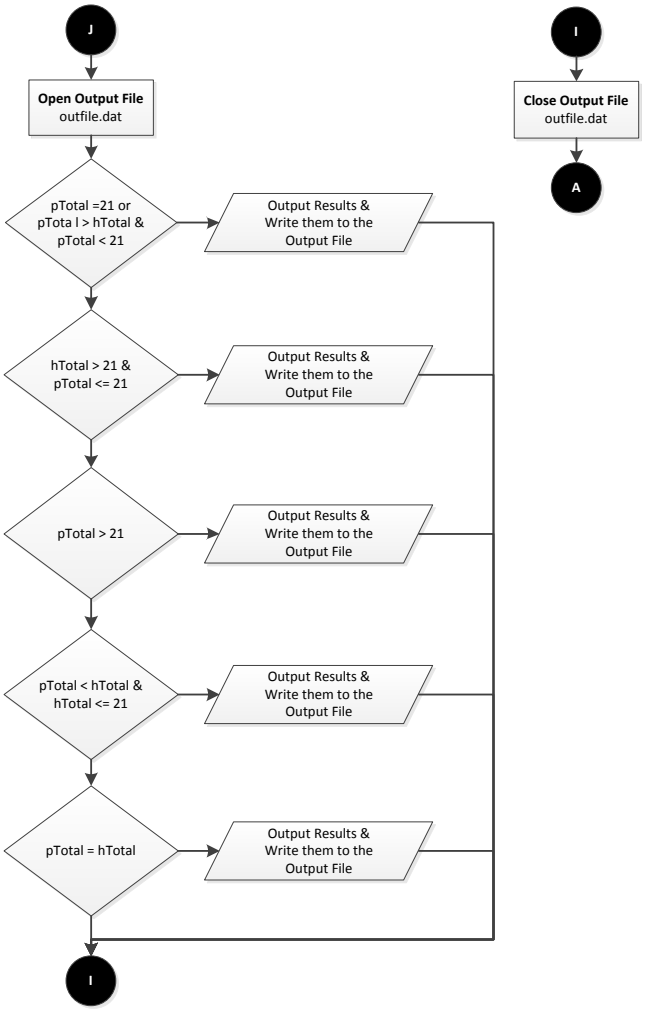
	outfile2	Variable used to identify file that is being read from
Character	ans	Input option to allow continue of play
Boolean	exitMnu	Alternative option to end program at menu selection

Flowchart




P
Open Output File
outfile2.dat
Input Name &
Write it to the File
Close Output File
outfile2.dat
Open Input File
infile.dat
Output Message
from File
Close Input File
infile.dat
Function Call (First Card)
dealCrd(card,randCard,suit,randSuit);
prntCrd(card,randCard,suit,randSuit);
calc(randCard, valu);
Ace Re-Adjustment
valu=1
YES
value=11
Q
Q
Calculate Player's Running Total
pTotal=rTotal+valu;
rTotal=pTotal;
Function Call (Second Card)
dealCrd(card,randCard,suit,randSuit);
prntCrd(card,randCard,suit,randSuit);
calc(randCard, valu);
Ace Re-Adjustment for Additional Cards
rTotal>10 &
valu=11
YES
value=1
rTotal<=10 &
valu=1
YES
value=11
Calculate Player's Total
pTotal=rTotal+valu;
Output Player's
Total
R
R
Player Total < 21
NO
H
Would you Like
Another Card?
Function Call (Third Card)
dealCrd(card,randCard,suit,randSuit);
prntCrd(card,randCard,suit,randSuit);
calc(randCard, valu);
rTotal>10 &
valu=11
YES
value=1
rTotal<=10 &
valu=1
YES
value=11
Calculate Player's Total
pTotal=rTotal+valu;
Output Player's
Total
S
S
Player Total < 21
NO
H
Would you Like
Another Card?
Function Call (Fourth Card)
dealCrd(card,randCard,suit,randSuit);
prntCrd(card,randCard,suit,randSuit);
calc(randCard, valu);
rTotal>10 &
valu=11
YES
value=1
rTotal<=10 &
valu=1
YES
value=11
Calculate Player's Total
pTotal=rTotal+valu;
Output Player's
Total
T


T
Player Total < 21
NO
H
Would you Like Another Card?
Function Call (Fifth Card)
dealCrd(card,randCard,suit,randSuit);
prntCrd(card,randCard,suit,randSuit);
calc(randCard, valu);
rTotal>10 & valu=11
YES
value=1
rTotal<=10 & valu=1
YES
value=11
Calculate Player's Total
pTotal=rTotal+valu;
Output Player's Total
U
U
Player Total < 21
NO
H
Would you Like Another Card?
Function Call (Sixth Card)
dealCrd(card,randCard,suit,randSuit);
prntCrd(card,randCard,suit,randSuit);
calc(randCard, valu);
rTotal>10 & valu=11
YES
value=1
rTotal<=10 & valu=1
YES
value=11
Calculate Player's Total
pTotal=rTotal+valu;
Output Player's Total
H
H
Function Call (Dealer's First Card)
dealCrd(card,randCard,suit,randSuit);
prntCrd(card,randCard,suit,randSuit);
calc(randCard, valu);
Ace Re-Adjustment for Additional Cards
hTotal > 10 & valu = 11
value=1
hTotal <= 10 & valu = 1
value=11
Calculate The House's Total
hTotal=hrTot+valu;
hrTot=hTotal;
Function Call (Dealer's Additional Card)
dealCrd(card,randCard,suit,randSuit);
prntCrd(card,randCard,suit,randSuit);
calc(randCard, valu);
Ace Re-Adjustment for Additional Cards
hTotal > 10 & valu = 11
value=1
hTotal <= 10 & valu = 1
value=11
I
I
Calculate House's Total
hTotal=hTotal+valu;
hTotal < 16
Output Player's Total
J


J
Open Output File
outfile.dat
pTotal =21 or
pTota l > hTotal &
pTotal < 21
Output Results &
Write them to the
Output File
hTotal > 21 &
pTotal <= 21
Output Results &
Write them to the
Output File
pTotal > 21
Output Results &
Write them to the
Output File
pTotal < hTotal &
hTotal <= 21
Output Results &
Write them to the
Output File
pTotal = hTotal
Output Results &
Write them to the
Output File
I
I
Close Output File
outfile.dat
A

Program Code

```
/*
 * File:   main.cpp
 * Author: Victor Medel
 * Created on June 1, 2014, 9:30 PM
 * CSC 5 (42450) | Project 2: Black Jack Game Version 2
 */

//System Libraries
#include <iostream>
#include <ctime>
#include <cstdlib>
#include <string>
#include <iomanip>
#include <fstream>
using namespace std;

//Global Constants

//Function Prototypes
void dealCrd(string card[], int& randCard, string suit[], int& randSuit);
void prntCrd(string card[], int& randCard, string suit[], int& randSuit);
int calc(int& randCard, int& valu);

//Execution Starts Here
int main(int argc, char** argv) {
    //Variable Declaration and Random Seed
    srand(time(0));
    int valu, pTotal, rTotal, hTotal, hrTot;
    int randCard, randSuit;
    string suit[4]={"of Hearts","of Diamonds","of Spades","of Clubs"};
    string card[14]={"Ace","2","3","4","5","6","7","8","9","10","Ace","Jack","Queen","King"};
    char ans;
    ifstream infile;
    ofstream outfile,outfile2;
    int choice;
    bool exitMnu=true;
    string next,name;
    //Loop until exit
    do{
        //Output Menu
        cout<<"\n";
        cout<<"Select From The Menu"<<endl;
        cout<<"\n";
        cout<<"1. Play Blackjack"<<endl;
        cout<<"2. Blackjack Game Overview"<<endl;
        cout<<"***Anything Else Exit Program***"<<endl;
        cout<<"\n";
        //Input your choice
        cout<<"Selection: ";
        cin>>choice;
        //Solve the problem chosen
        switch(choice){
            case 1:
    //Initialize Player and House Running Totals
    hrTot=0;
    rTotal=0;
```

```
57  //Open File to write name
58                  outfile2.open("output2.dat", ios::app);
59                     if(outfile2.fail()){
60                        cout<<"Input file failed to open.\n";
61                        exit(1);
62                     }
63                  cout<<"Please enter your first name: ";
64                  cin>>name;
65                  outfile2<<name<<" ";
66                  outfile2.close();
67  //Open File to read welcome message
68                  infile.open("input.dat");
69                     if(infile.fail()){
70                        cout<<"Input file failed to open.\n";
71                        exit(1);
72                     }
73                  cout<<name<<" ";
74                  while (infile>>next){
75                     cout<<next<<" ";
76                  }
77                  infile.close();
78                  cout<<endl<<endl;
79  //Function Call
80  dealCrd(card,randCard,suit,randSuit);
81  prntCrd(card,randCard,suit,randSuit);
82  calc(randCard, valu);
83  //Ace Re-Adjustment
84  if(valu==1){
85     valu=11;
86  }
87  //End of Ace Re-Adjustment
88     pTotal=rTotal+valu;
89     rTotal=pTotal;
90  cout<<" | ";
91  dealCrd(card,randCard,suit,randSuit);
92  prntCrd(card,randCard,suit,randSuit);
93  calc(randCard, valu);
94  //Ace Re-Adjustment for Additional Cards
95  if(rTotal>10&&valu==11){
96     valu=1;
97  }else if(rTotal<=10&&valu==1){
98     valu=11;
99  }
100  //End of Ace Re-Adjustment
101     pTotal=rTotal+valu;
102     cout<<"\n\n";
103     cout<<"Your score is: ";
104     cout<<pTotal;
105     cout<<"\n";
106                  //Option to Allow Player to Hit and Continue Playing
107                  //Three additional cards always exceed a score of 21
108                  if(pTotal<21){
109                   cout<<"Would you like another card?\n";
110                   cout<<"Enter y for yes, anything else for no: ";
111                   cin>>ans;
112                     if (ans=='y'||ans=='Y'){
113                        //Players Additional Cards
114                        cout<<"\n";
115                        cout<<"You have been dealt a ";
116                        dealCrd(card,randCard,suit,randSuit);
117                        prntCrd(card,randCard,suit,randSuit);
118                        calc(randCard, valu);
119                        //Ace Re-Adjustment for Additional Cards
120                         if(pTotal>10&&valu==11){
```

```
121                     valu=1;
122                 }else if(pTotal<=10&&valu==1){
123                     valu=11;
124                 }
125                 //End of Ace Re-Adjustment
126                 pTotal=pTotal+valu;
127                 cout<<"\n";
128                 cout<<"Your score is now: ";
129                 cout<<pTotal;
130                 cout<<"\n\n";
131                     if (pTotal<21){
132                         cout<<"Would you like another card?\n";
133                         cout<<"Enter y for yes, anything else for no: ";
134                         cin>>ans;
135                         if (ans=='y'||ans=='Y'){
136                             //Players Additional Card
137                             cout<<"\n";
138                             cout<<"You have been dealt a ";
139                             dealCrd(card,randCard,suit,randSuit);
140                             prntCrd(card,randCard,suit,randSuit);
141                             calc(randCard, valu);
142                             //Ace Re-Adjustment for Additional Cards
143                             if(pTotal>10&&valu==11){
144                                 valu=1;
145                             }else if(pTotal<=10&&valu==1){
146                                 valu=11;
147                             }
148                             //End of Ace Re-Adjustment
149                             pTotal=pTotal+valu;
150                 cout<<"\n";
151                 cout<<"Your score is now: ";
152                 cout<<pTotal;
153                 cout<<"\n\n";
154                 if (pTotal<21){
155                     cout<<"Would you like another card?\n";
156                     cout<<"Enter y for yes, anything else for no: ";
157                     cin>>ans;
158                     if (ans=='y'||ans=='Y'){
159                         //Players Additional Card
160                         cout<<"\n";
161                         cout<<"You have been dealt a ";
162                         dealCrd(card,randCard,suit,randSuit);
163                         prntCrd(card,randCard,suit,randSuit);
164                         calc(randCard, valu);
165                         //Ace Re-Adjustment for Additional Cards
166                         if(pTotal>10&&valu==11){
167                             valu=1;
168                         }else if(pTotal<=10&&valu==1){
169                             valu=11;
170                         }
171                         //End of Ace Re-Adjustment
172                         pTotal=pTotal+valu;
173                 cout<<"\n";
174                 cout<<"Your score is now: ";
175                 cout<<pTotal;
176                 cout<<"\n\n";
177                 //Fourth Card Option for the Risk Taker
178                 if(pTotal<21){
179                     cout<<"Would you like another card?\n";
180                     cout<<"Enter y for yes, anything else for no: ";
181                     cin>>ans;
182                     if (ans=='y'||ans=='Y'){
183                         //Players Additional Card
184                         cout<<"\n";
```

```
                    cout<<"You have been dealt a ";
                    dealCrd(card,randCard,suit,randSuit);
                    prntCrd(card,randCard,suit,randSuit);
                    calc(randCard, valu);
                    //Ace Re-Adjustment for Additional Cards
                     if(pTotal>10&&valu==11){
                        valu=1;
                     }else if(pTotal<=10&&valu==1){
                        valu=11;
                     }
                     //End of Ace Re-Adjustment
                    pTotal=pTotal+valu;
                  cout<<"\n";
                  cout<<"Your score is now: ";
                  cout<<pTotal;
                  cout<<"\n\n";
                    }
                  }
                 }
                }
                 }
                }
            }else;
        }

        //House's Hand
        cout<<"\n";
        cout<<"The house has been dealt the following cards: ";
        cout<<"\n";
              dealCrd(card,randCard,suit,randSuit);
              prntCrd(card,randCard,suit,randSuit);
              calc(randCard, valu);
              //Ace Re-Adjustment for Additional Cards
               if(hTotal>10&&valu==11){
                  valu=1;
               }else if(hTotal<=10&&valu==1){
                  valu=11;
               }
               //End of Ace Re-Adjustment
        hTotal=hrTot+valu;
        hrTot=hTotal;
        do{
        cout<<" | ";
              dealCrd(card,randCard,suit,randSuit);
              prntCrd(card,randCard,suit,randSuit);
              calc(randCard, valu);
              //Ace Re-Adjustment for Additional Cards
               if(hTotal>10&&valu==11){
                  valu=1;
               }else if(hTotal<=10&&valu==1){
                  valu=11;
               }
               //End of Ace Re-Adjustment
        hTotal=hTotal+valu;
        //Based on Blackjack Rules House continues to deal
        //itself a card if total score is less than 16
        }while(hTotal<16);
        cout<<"\n";
        cout<<"The house's score is: ";
        cout<<hTotal;
        cout<<"\n";
        //Outcome Output
        //Open File to write results
         outfile.open("output.dat", ios::app);
```

```
                    if(outfile.fail()){
                      cout<<"Input file failed to open.\n";
                      exit(1);
                    }
                  if(pTotal==21||(pTotal>hTotal&&pTotal<21)){
                    cout<<"\n";
                    cout<<"***Congratulations! You have won***";
                    outfile<<"Win"<<" "<<endl;
                    cout<<"\n";
                  }else if(hTotal>21&&pTotal<=21) {
                    cout<<"\n";
                    cout<<"***Congratulations! You have won***";
                    outfile<<"Win"<<" "<<endl;
                    cout<<"\n";
                  }else if(pTotal>21){
                    cout<<"\n";
                    cout<<"***Bust***";
                    outfile<<"Loss"<<" "<<endl;
                    cout<<"\n";
                  }else if(pTotal<hTotal&&hTotal<=21){
                    cout<<"\n";
                    cout<<"***House Wins***";
                    outfile<<"Loss"<<" "<<endl;
                    cout<<"\n";
                  }else if(pTotal==hTotal){
                    cout<<"\n";
                    cout<<"***Stand-Off/Draw, Play Again***";
                    outfile<<"Tie"<<" "<<endl;
                    cout<<"\n";
                  }
                   outfile.close();
                   //Exit Stage Right
                        break;
                        case 2:
                           //Objective of my game
                       cout<<"\n";
                        cout<<"The object of the game is to beat the house \n";
                        cout<<"by receiving a score of 21 or by getting a higher \n";
                        cout<<"score than the house without going over 21 with\n";
                        cout<<"any additional cards. The game begins by dealing\n";
                        cout<<"two cards to the player; after displaying your\n";
                        cout<<"score  and if your score is less than 21 you \n";
                        cout<<"will have the option to take another\n";
                        cout<<"card to add to your total score or hold with\n";
                        cout<<"your existing score. If you hold or go over 21\n";
                        cout<<"after choosing another card the program will \n";
                        cout<<"automatically display the house's hand and then\n";
                        cout<<"determine the outcome.\n\n";
                        cout<<"Multiple decks of cards are used with the following values:\n\n";
                        cout<<"Cards 2 through 10 = face value points\n";
                        cout<<"Jacks = 10 points\n";
                        cout<<"Queens = 10 Points\n";
                        cout<<"Kings = 10 Points\n";
                        cout<<"Aces = 1 or 11 depends on the player's total. If player's total \n";
                        cout<<"is less than ten points then they hold a value of eleven otherwise \n";
                        cout<<"the card will hold a value of one.\n";
                        ;break;
                        default: exitMnu=false;
                      }
                   }while(exitMnu);
                   //Exit Stage Right
    return 0;
}
void dealCrd(string card[], int& randCard, string suit[], int& randSuit){
```

```
313     //Randomly Selects Card and Suit
314     randSuit=rand()%4;
315     randCard=rand()%14;
316     card[randCard];
317     suit[randSuit];
318 }
319 //Prints Card
320 void prntCrd(string card[], int& randCard, string suit[], int& randSuit){
321     //Outputs One Card
322     cout<<right<<setw(2)<<card[randCard];
323     cout<<" ";
324     cout<<left<<setw(2)<<suit[randSuit];
325 }
326 //Function Calculates the Value of each card
327 int calc(int& randCard, int& valu){
328     //Assigns Value to Card Dealt
329      if(randCard==0){
330         valu=1;
331     }else if(randCard==1){
332         valu=2;
333     }else if(randCard==2){
334         valu=3;
335     }else if(randCard==3){
336         valu=4;
337     }else if(randCard==4){
338         valu=5;
339     }else if(randCard==5){
340         valu=6;
341     }else if(randCard==6){
342         valu=7;
343     }else if(randCard==7){
344         valu=8;
345     }else if(randCard==8){
346         valu=9;
347     }else if(randCard==9){
348         valu=10;
349     }else if(randCard==10){
350         valu=11;
351     }else if(randCard==11){
352         valu=10;
353     }else if(randCard==12){
354         valu=10;
355     }else if(randCard==13){
356         valu=10;
357     }
358      return valu;
359 }
360 /*
361  * I attempted to use 2D array but it was not functional with the Blackjack
362  * game I was developing. I've included the code to demonstrated my attempt.
363  *
364  void dealCrd(int card[][COLS], int n, int randSuit){
365     for(int i=0;i<n;i++){
366         card[i][0]=rand()%14+1;
367         card[i][1]=suit(randSuit);
368     }
369 }
370
371 void prntCrd(int card[][COLS], int n, int randSuit){
372     for (int i=0;i<n;i++){
373        cout<<card[i][0];
374        cout<<" of ";
375         if(card[i][0]==0){
376             cout<<"Hearts";
```

```
        }else if(card[i][0]==1){
           cout<<"Diamonds";
        }else if(card[i][0]==2){
           cout<<"Clubs";
        }else if(card[i][0]==3){
           cout<<"Spades";
        }
    }
}
int suit(int& randSuit){
    randSuit=rand()%4+1;
    return randSuit;
}
 */
```