

Blackjack_V1.1

Card Game

Shifeng Song
CSC5 Summer 2017
45561

TABLE OF CONTENTS

INTRODUCTION -----	3
HOW THE CARD GAME WORKS -----	3
OBJECT OF THE GAME-----	3
HOW TO PLAY-----	3
FLOWCHART-----	4
CROSS REFERENCE -----	12
PROOF OF A WORKING PRODUCT -----	14
CODE -----	18

Introduction

Blackjack is a fast pace casino game. It is part of table and card games offered in every casino around the world and it is the second most popular card gambling game, after poker. The enormous popularity of blackjack is obvious once you understand that blackjack is one of the few gambling games that are not solely reliant on luck. This means that skilled players that master basic blackjack strategy can gain an advantage over the casino.

How the Card Game Works

Object of the Game

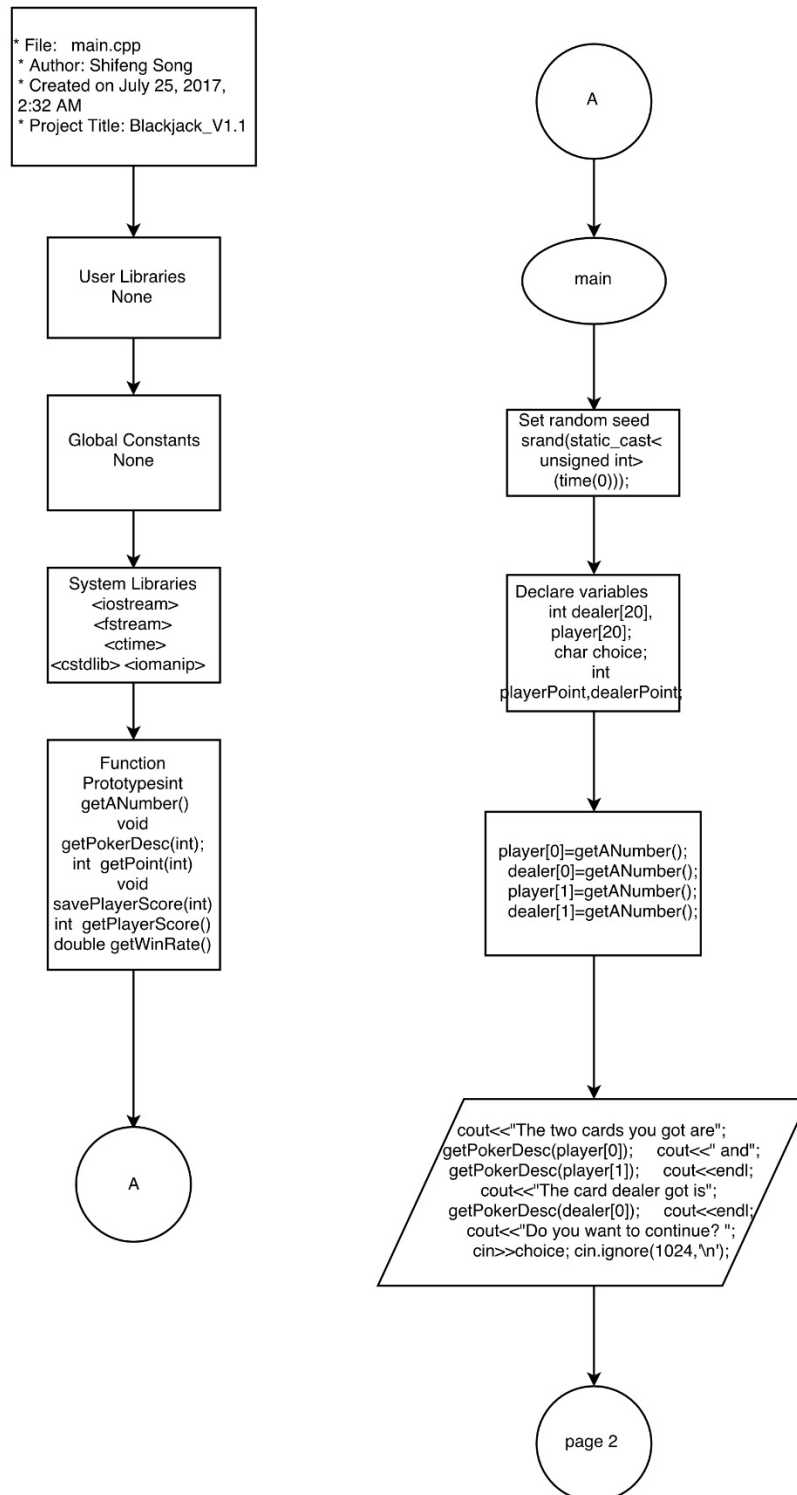
A traditional 52 cards deck is used (or several decks) and each card has a point value attached to it. Face cards (King, Queen and Jack) count as 10 and all other cards keep the value that is printed on them.

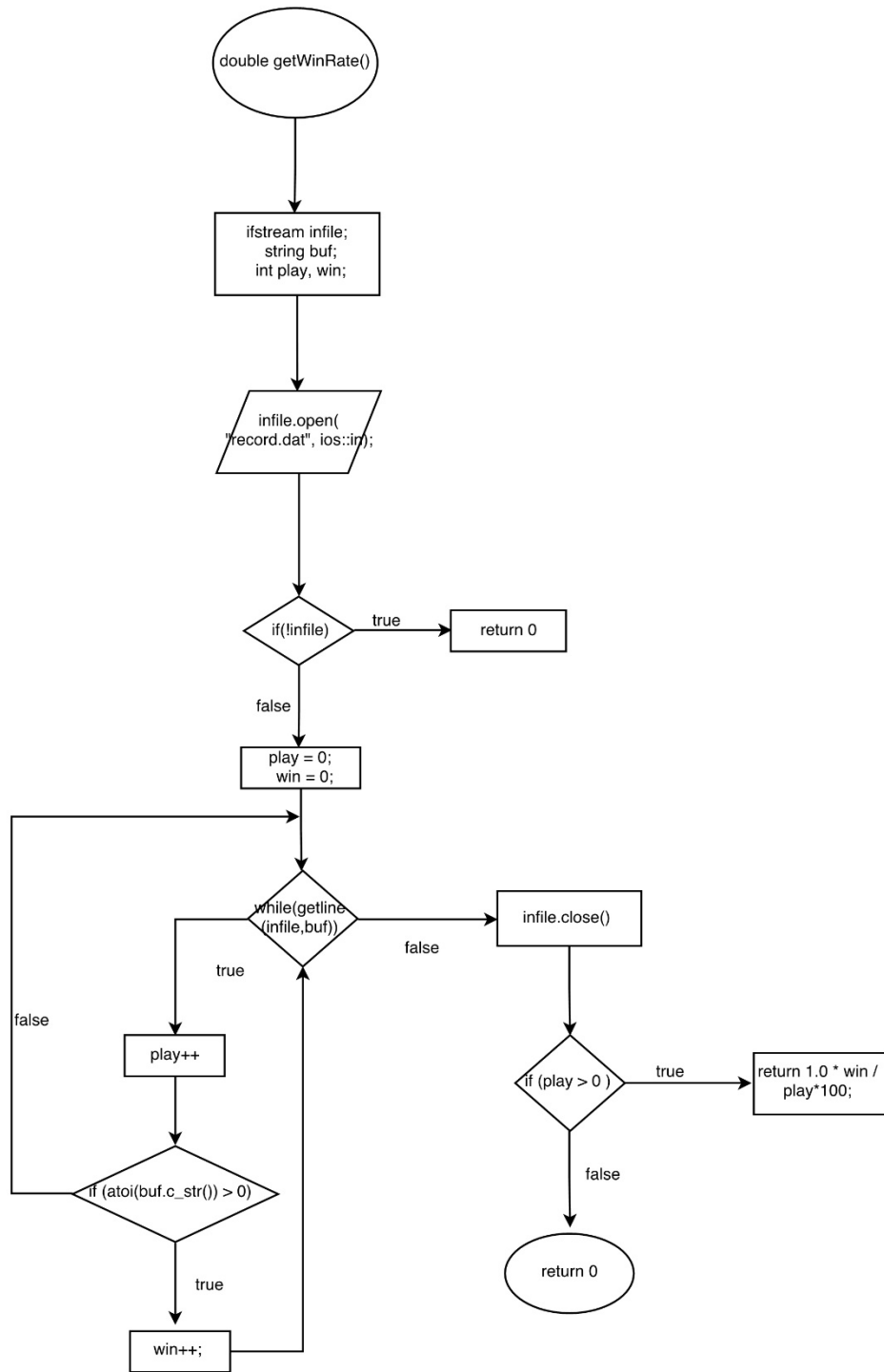
How to play

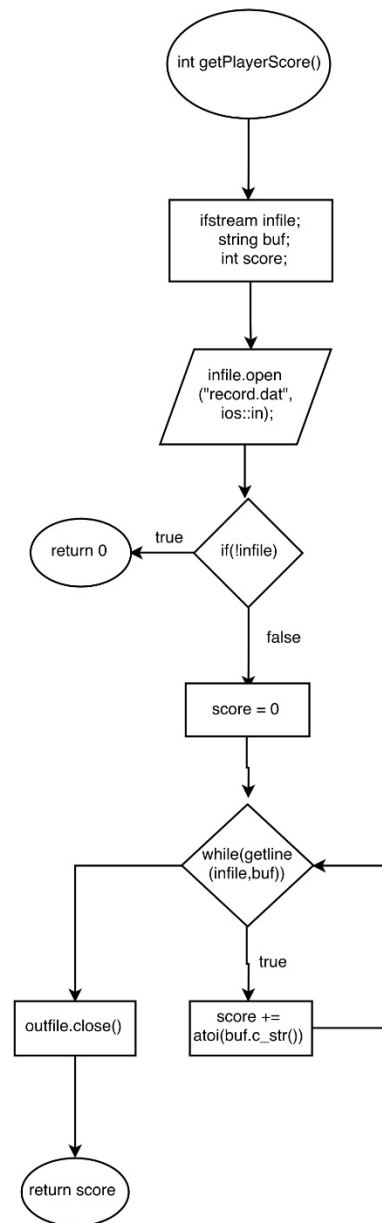
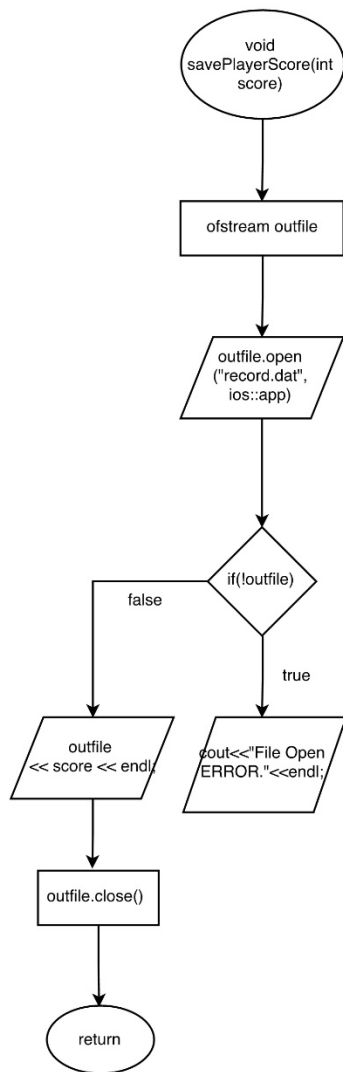
The dealer deals each player two cards facing up. The dealer also gets two cards but only one is facing up and the other is facing down (known as the hole card). Players are allowed to draw additional cards to the total hand value of 21. Once a player drew a card that takes his total hand value above 21 they bust out of the game. Both the dealer or player can win the hand with low valued hands in situations when one or the other busted out of the game.

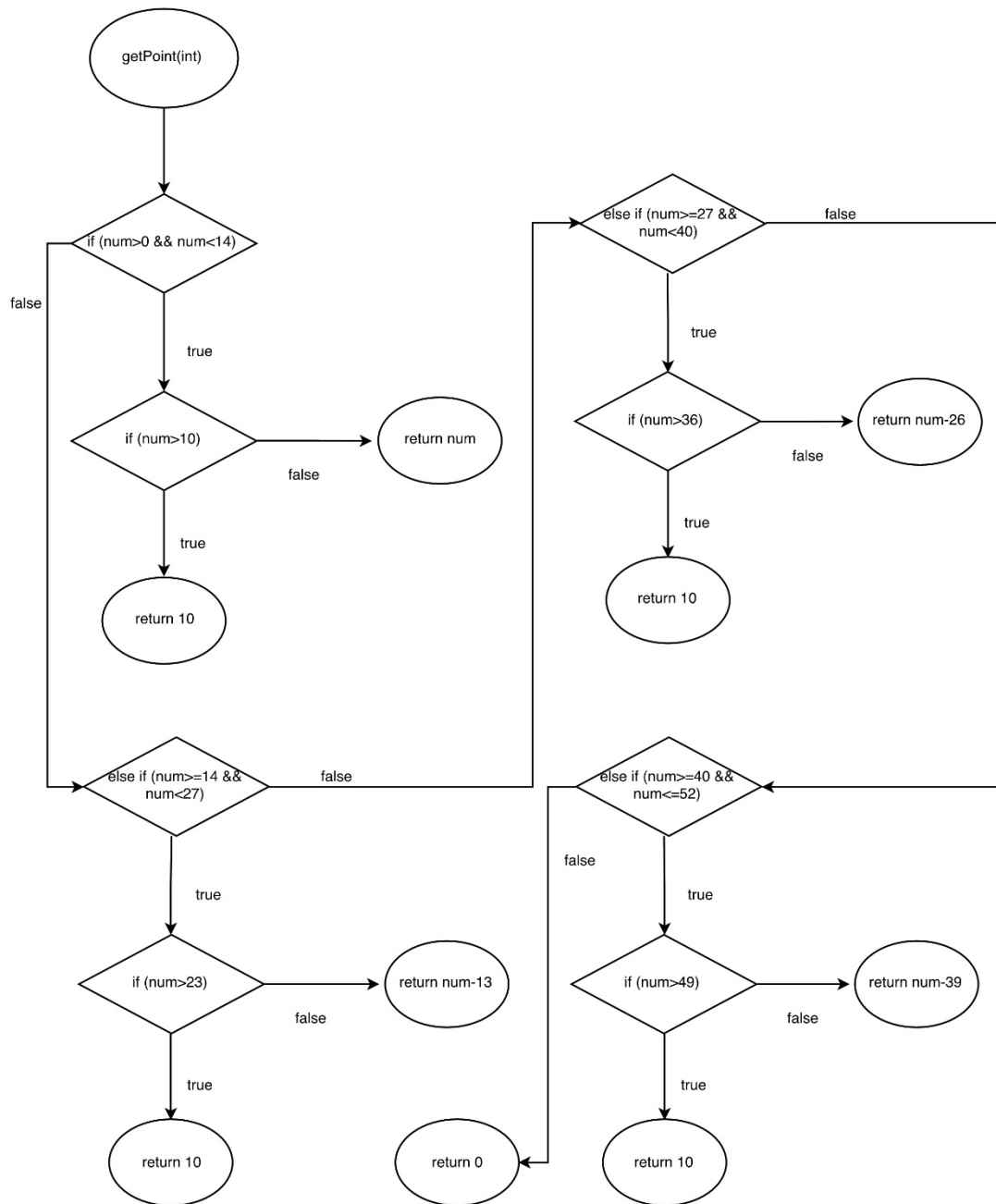
Flowchart

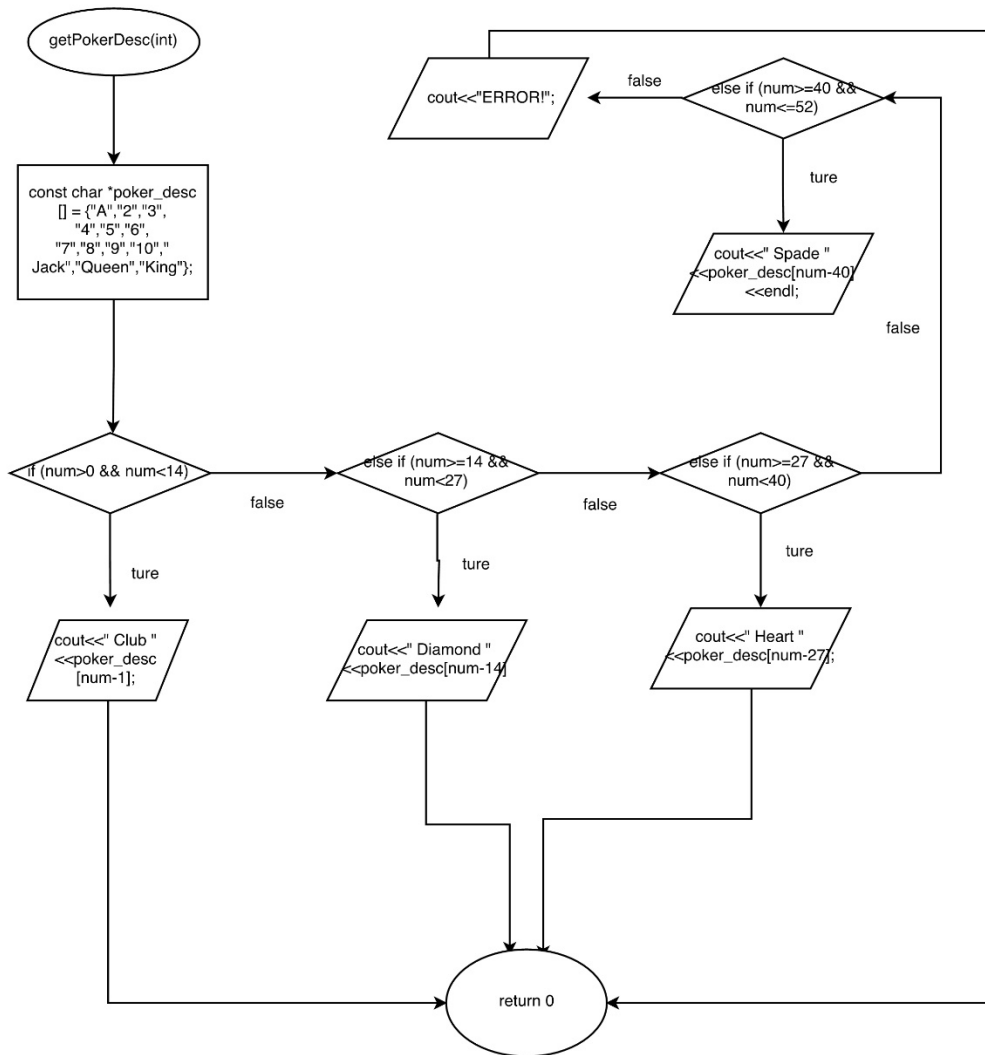
Project_2_BlackJack_V1.1_Flowchart

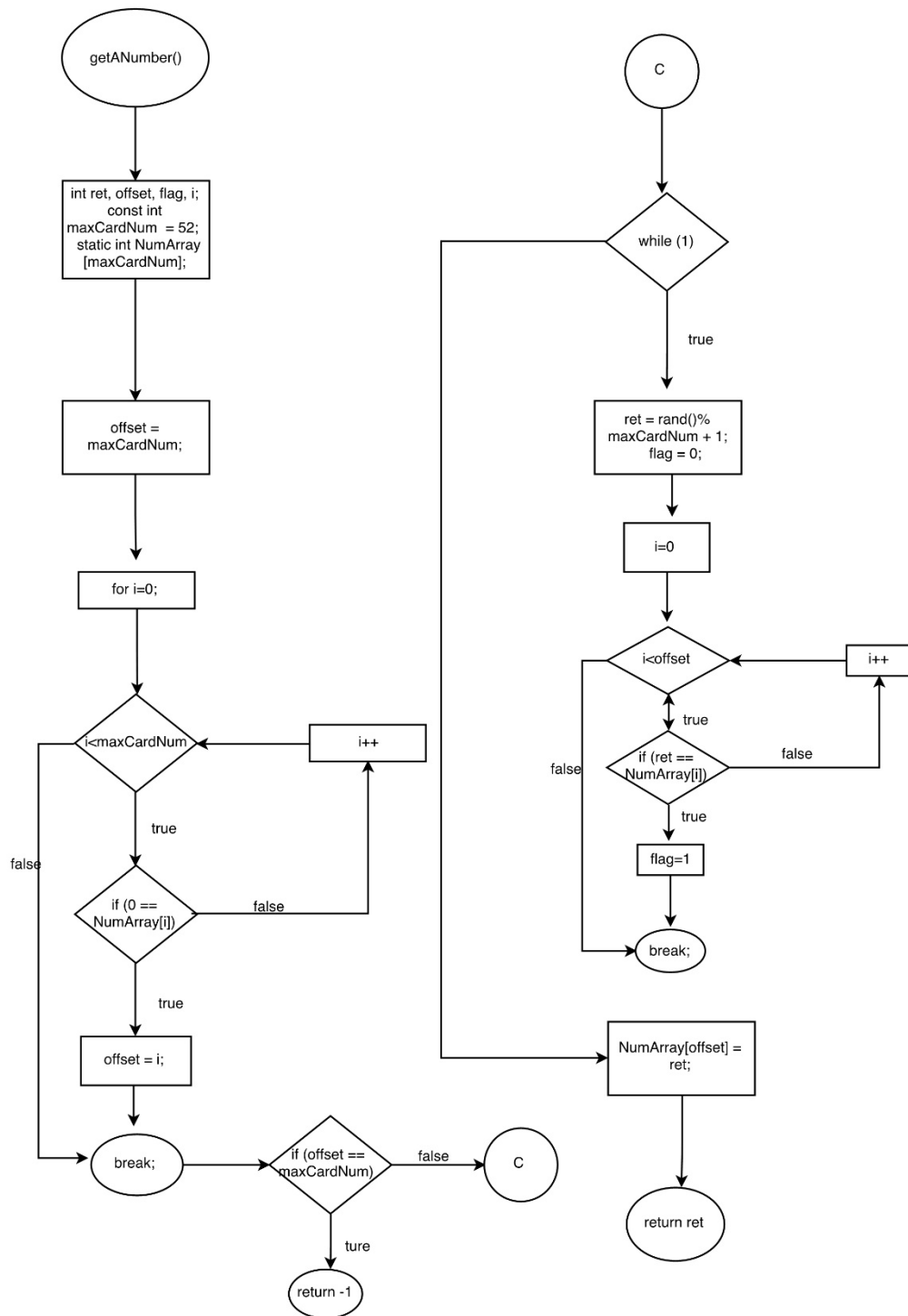


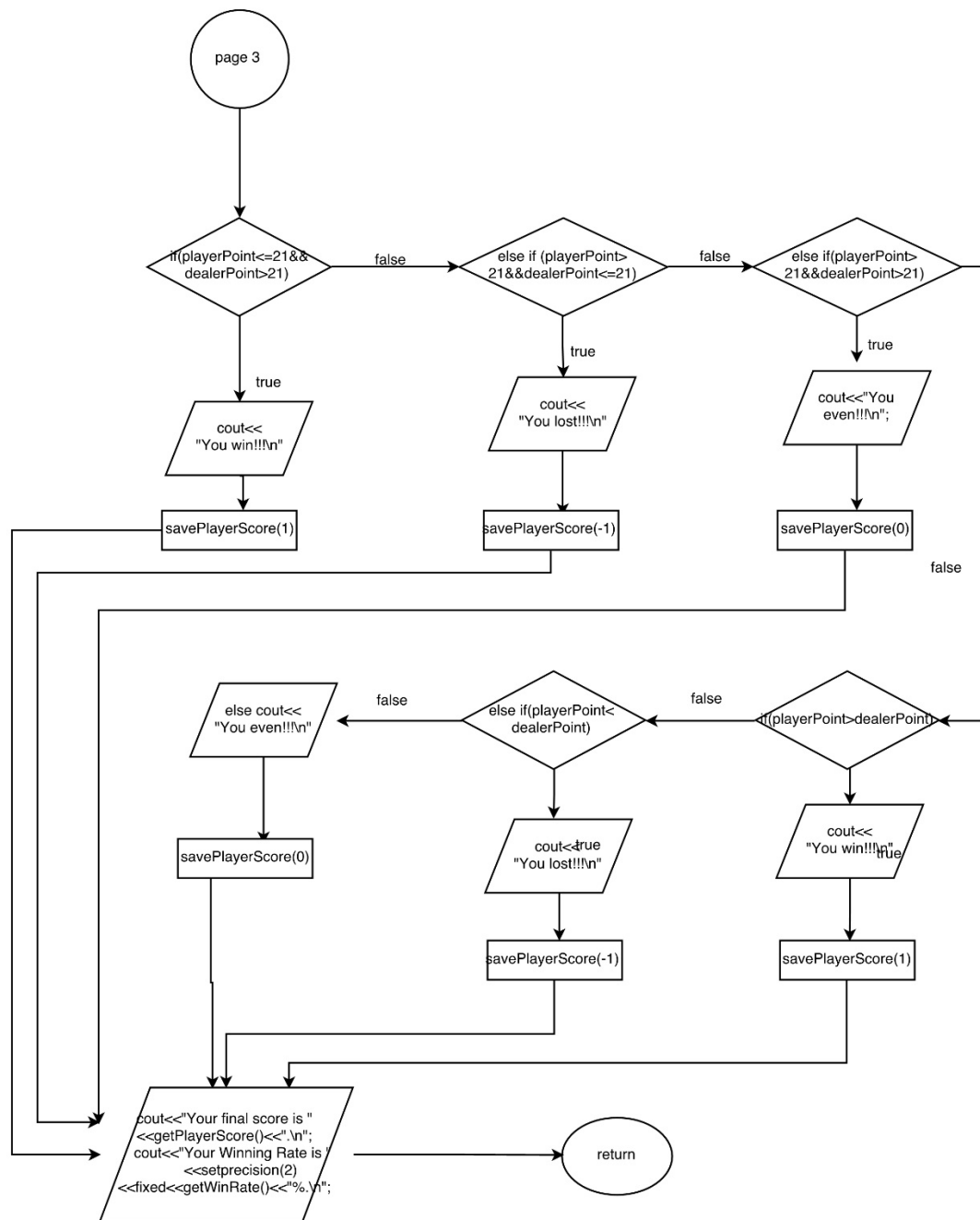


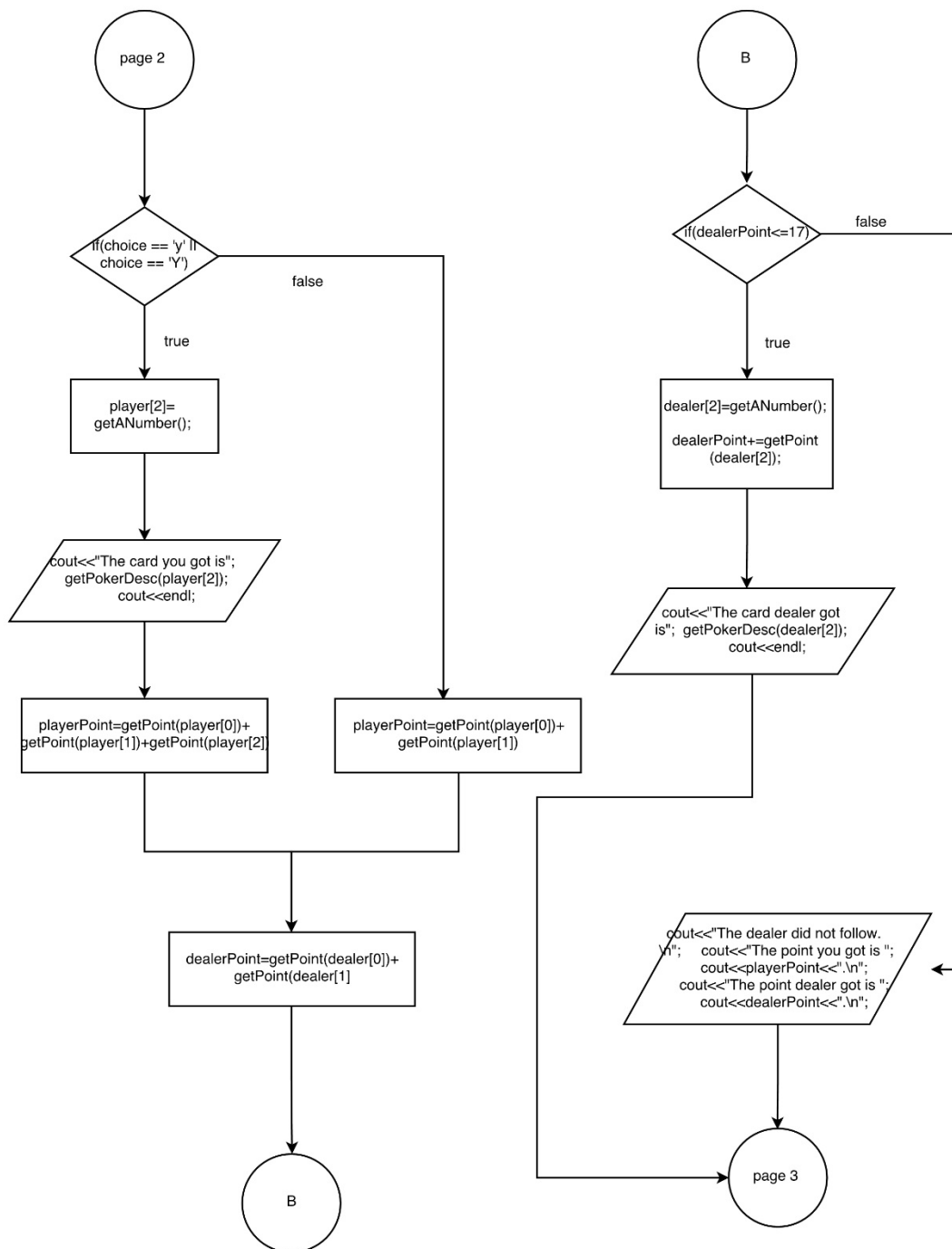












Cross Reference

Chapter	Section	Topic	Where in Code Line number
2	2	cout	49, 51, 53, 55, 57, 59, 65, 67, 78, 80, 83, 85, 87, 88,89,91,92,95,96,97,101,102,103,107,108, 109,114,115,116,120,121,122,126,127,128,133,134
	3	libraries	iostream, iomanip,cstdlib, fstream, ctime
	4	variables/literals	37,38,39
	5	Identifiers	
	6	Integers	34,37,39,146,148,149,150,188,212,243,256,259,277
	7	Characters	38, 189,
	8	Strings	258, 276
	9	Floats No Doubles	274,
	10	Bools	none
	11	Sizeof *****	none
	12	Variables 7 characters or less	none
	13	Scope ***** No Global Variables	none
	14	Arithmetic operators	none
	15	Comments 20%+	141-145,181-187, 205-211,239-241
	16	Named Constants	149, 189
	17	Programming Style ***** Emulate	none
3	1	cin	59,60
	2	Math Expression	none
	3	Mixing data types ****	none
	4	Overflow/Underflow *****	none
	5	Type Casting	34
	6	Multiple assignment *****	none
	7	Formatting output	134
	8	Strings	258, 276
	9	Math Library	none
	10	Hand tracing *****	none
4	1	Relational Operators	74,93, 99, 105, 112,118,165,192,194,196,198,213, 214,219,220,224,225,229,230,286,291,
	2	if	153,159,166,172,262,280,286,

	4	If-else	245-249,291-293
	5	Nesting	93-124,213-234
	6	If-else-if	192-200
	7	Flags *****	none
	8	Logical operators	93,99,105,192,194,196,198,213,219,224,229,62, 247,262,280
	11	Validating user input	62
	13	Conditional Operator	none
	14	Switch	none
5	1	Increment/Decrement	152,165,285,286
	2	While	161,264,284
	5	Do-while	none
	6	For loop	152,165
	11	Files input/output both	244-268
	12	No breaks in loops *****	none
6	3	Function Prototypes	25,26,27,28,29,30
	5	Passing by value	49,51,55,65,78,85,67,70,73,76,97,103,109,116, 122,128
	8	Returning values from functions	42-45,63,75,132,134
	9	Returning a boolean *****	<i>none</i>
	10	No Global Variables Allowed	none
		Only Global Constants	none
		Meaning Conversions,Physical Constants only	none
	11	Static Local	150
	12	Default arguments	none
	13	Reference Parameters	none
	14	Overloading functions	none
	15	Exit function *****	none
7	4	Array Initialization	189,190
	6	Processing Arrays	none

	7	Parallel Arrays	none
	8	Arrays as function arguments	none
	9	2-D Arrays	none
	12	STL Vector	none
8	1	Linear and Binary Search	146-178
	3	Bubble and Selection Sort	none
	5	Search/Sorting Vectors *****	none

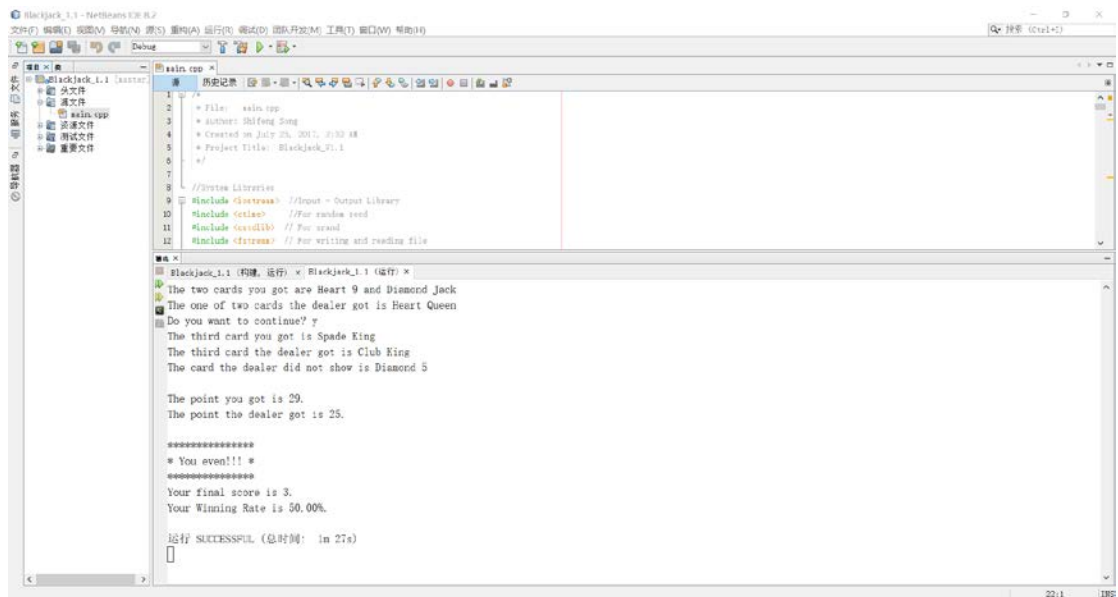
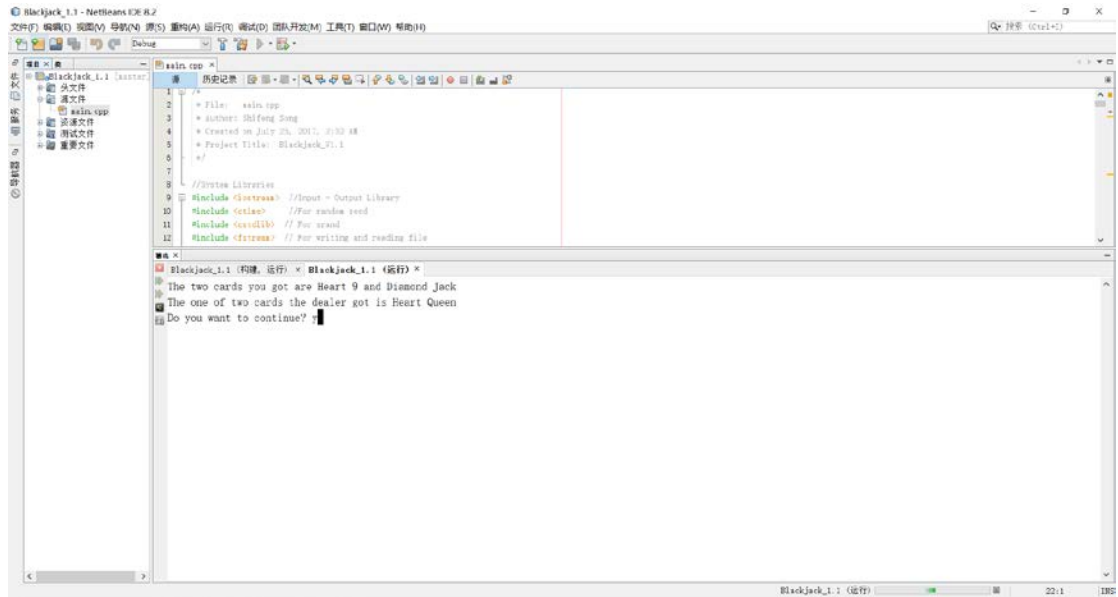
Proof of a Working Product

I have provided some screenshots that prove that the program did work at one time on the next few pages.

```

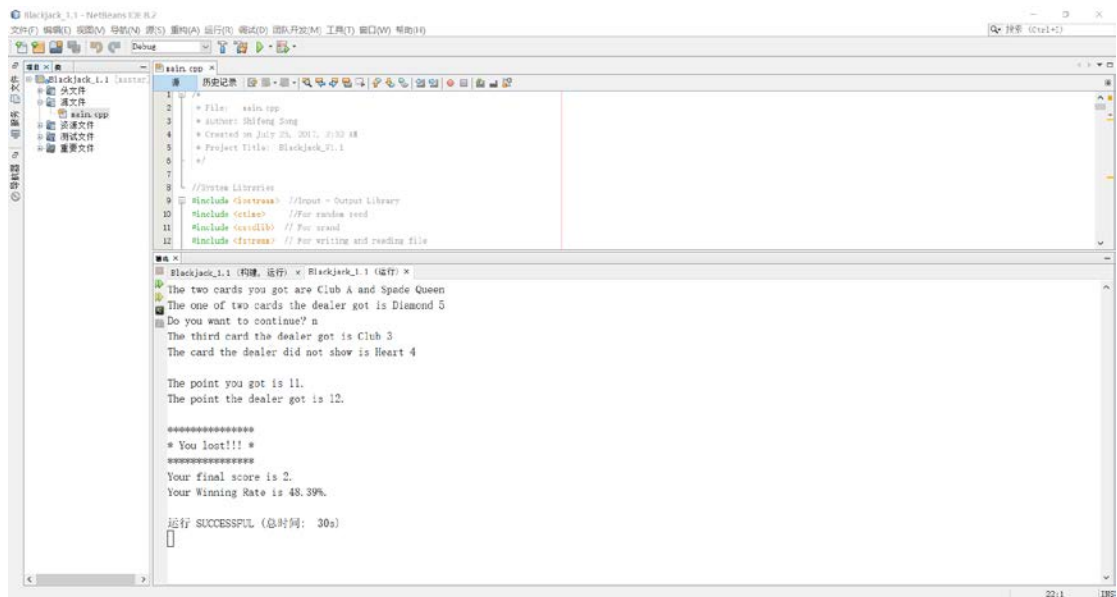
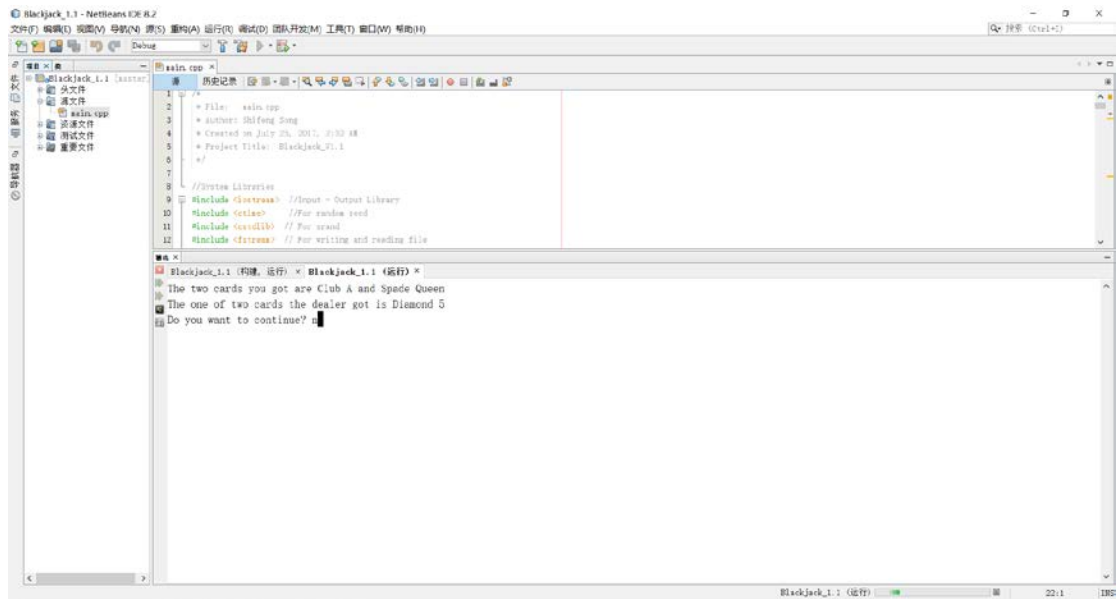
1 //
2 * File: main.cpp
3 * Author: Shifeng Song
4 * Created on July 25, 2017, 2:02:48
5 * Project Title: Blackjack_1.1
6 *
7
8 //Include Libraries
9 #include <iostream> //Input - Output library
10 #include <vector> //For random seed
11 #include <cstdlib> //For srand
12 #include <iostream> //For writing and reading file
13
14 int main()
15 {
16     //The two cards you got are Heart 9 and Diamond Jack
17     //The one of two cards the dealer got is Heart Queen
18     //Do you want to continue?
19 }

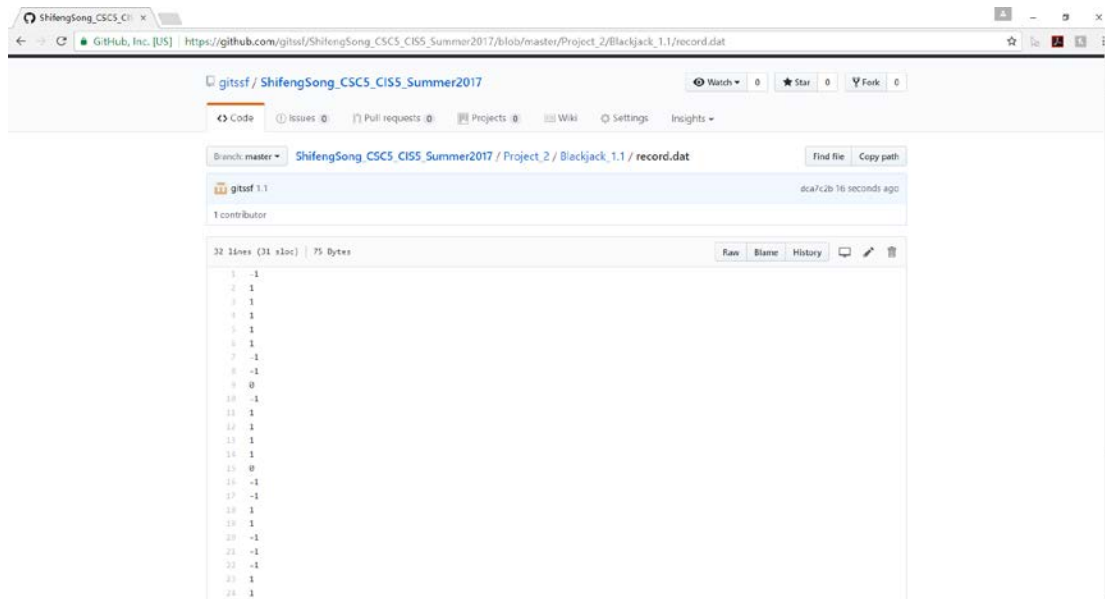
```



```
1 //
2 * File: main.cpp
3 * Author: Shifeng Song
4 * Created on July 25, 2017, 2:32 AM
5 * Project Title: Blackjack_V1.1
6 *
7
8 //Include Libraries
9 #include <iostream> //Input - Output library
10 #include <ctime> //For random seed
11 #include <cstdlib> //For srand
12 #include <fstream> //For writing and reading file
13
14
15 Blackjack_V1.1 (构建: 运行) x Blackjack_V1.1 (运行) x
16 The two cards you got are Spade 10 and Heart 2
17 The one of two cards the dealer got is Heart Jack
18 Do you want to continue? y
19 The third card you got is Heart 5
20 The third card the dealer got is Heart 9
21 The card the dealer did not show is Club 6
22
23 The point you got is 17.
24 The point the dealer got is 25.
25
26 *****
27 * You win!!! *
28 *****
29 Your final score is 4.
30 Your Winning Rate is 51.72%.
31
32 运行 SUCCESSFUL (总时间: 2s)
```

```
1 //
2 * File: main.cpp
3 * Author: Shifeng Song
4 * Created on July 25, 2017, 2:32 AM
5 * Project Title: Blackjack_V1.1
6 *
7
8 //Include Libraries
9 #include <iostream> //Input - Output library
10 #include <ctime> //For random seed
11 #include <cstdlib> //For srand
12 #include <fstream> //For writing and reading file
13
14
15 Blackjack_V1.1 (构建: 运行) x Blackjack_V1.1 (运行) x
16 The two cards you got are Heart A and Spade 2
17 The one of two cards the dealer got is Club 3
18 Do you want to continue? y
19 The third card you got is Heart King
20 The third card the dealer got is Spade 10
21 The card the dealer did not show is Heart 7
22
23 The point you got is 13.
24 The point the dealer got is 20.
25
26 *****
27 * You lost!!! *
28 *****
29 Your final score is 3.
30 Your Winning Rate is 50.00%.
31
32 运行 SUCCESSFUL (总时间: 828ms)
```



Code(297 lines)

```

/*
 * File:   main.cpp
 * Author: Shifeng Song
 * Created on July 25, 2017, 2:32 AM
 * Project Title: Blackjack_V1.1
 */

//System Libraries
#include <iostream> //Input - Output Library
#include <ctime>     //For random seed
#include <cstdlib>   // For srand
#include <fstream>   // For writing and reading file
#include <iomanip>    // For Format

using namespace std; //Name-space under which system libraries exist

```

```

//User Libraries

//Global Constants


//Function Prototypes
int  getANumber();           //Get random number 1-52
void getPokerDesc(int);     //Transfer card suits
int  getPoint(int);         //Transfer card points
void savePlayerScore(int);  //Input data to a file
int  getPlayerScore();      //Read data from a file
double getWinRate();        //Get Player's Winning Rate


//Execution begins here
int main(int argc, char** argv){
    //Set random seed
    srand(static_cast<unsigned int> (time(0)));

    //Declare variables
    int dealer[20], player[20];
    char choice;
    int playerPoint,dealerPoint;

    //Initialize some variables
    player[0]=getANumber();
    dealer[0]=getANumber();
    player[1]=getANumber();
    dealer[1]=getANumber();

    //Map inputs to outputs the transformed data
    cout<<"The two cards you got are";
    getPokerDesc(player[0]);
    cout<<" and";
    getPokerDesc(player[1]);
    cout<<endl;

    cout<<"The one of two cards the dealer got is";
    getPokerDesc(dealer[0]);

```

```

cout<<endl;

cout<<"Do you want to continue? ";
cin>>choice;
cin.ignore(1024, '\n');

if(choice == 'y' || choice == 'Y') {
    player[2]=getANumber();
    cout<<"The third card you got is";
    getPokerDesc(player[2]);
    cout<<endl;

playerPoint=getPoint(player[0])+getPoint(player[1])+getPoint(player[2]);
}
else{
    playerPoint=getPoint(player[0])+getPoint(player[1]);
}

dealerPoint=getPoint(dealer[0])+getPoint(dealer[1]);
if(dealerPoint<=17){           //The dealer must be get third card
    dealer[2]=getANumber();    //if his points no more than 17
    dealerPoint+=getPoint(dealer[2]);
    cout<<"The third card the dealer got is";
    getPokerDesc(dealer[2]);
    cout<<endl;
}
else
    cout<<"The dealer did not follow. \n";

cout<<"The card the dealer did not show is";
getPokerDesc(dealer[1]);
cout<<"\n\n";
cout<<"The point you got is ";
cout<<playerPoint<<".\n";

cout<<"The point the dealer got is ";
cout<<dealerPoint<<".\n\n";

```

```

if(playerPoint<=21&&dealerPoint>21){
    cout<<"*****\n";
    cout<<"* You win!!! *\n";
    cout<<"*****\n";
    savePlayerScore(1);
}
else if(playerPoint>21&&dealerPoint<=21){
    cout<<"*****\n";
    cout<<"* You lost!!! *\n";
    cout<<"*****\n";
    savePlayerScore(-1);
}
else if(playerPoint>21&&dealerPoint>21){
    cout<<"*****\n";
    cout<<"* You even!!! *\n";
    cout<<"*****\n";
    savePlayerScore(0);
}
else {
    if(playerPoint>dealerPoint){
        cout<<"*****\n";
        cout<<"* You win!!! *\n";
        cout<<"*****\n";
        savePlayerScore(1);
    }
    else if(playerPoint<dealerPoint){
        cout<<"*****\n";
        cout<<"* You lost!!! *\n";
        cout<<"*****\n";
        savePlayerScore(-1);
    }
    else {
        cout<<"*****\n";
        cout<<"* You even!!! *\n";
        cout<<"*****\n";
        savePlayerScore(0);
    }
}

cout<<"Your final score is "<<getPlayerScore()<<".\n";
cout<<"Your Winning Rate is "
    <<setprecision(2)<<fixed<<getWinRate()<<"%.\n";

```

```

        //Exit stage right!
        return 0;
    }

    /*
    ** get random number 1-52
    ** input the numbers to getNumArr
    ** if the function is called more than 52 times then return -1
    */
    int getANumber()
    {
        int ret, offset, flag, i;
        const int maxCardNum = 52; // The Max number of playing cards
        static int NumArray [maxCardNum]; // Avoid repeating random numbers
        offset = maxCardNum;
        for (i=0; i<maxCardNum; i++) {
            if (0 == NumArray[i]) {
                offset = i;
                break;
            }
        }

        if (offset == maxCardNum) return -1;

        while (1) {
            ret = rand()%maxCardNum + 1;

            flag = 0;
            for (i=0; i<offset; i++) {
                if (ret == NumArray[i]) {
                    flag = 1;
                    break;
                }
            }

            if (0 == flag) break;
        }
    }

```

```

    NumArray[offset] = ret;

    return ret;
}

/*
** Return card suits
** 1-13 Club A、2、3...10、J、Q、K
** 14-26 Diamond A、2、3...10、J、Q、K
** 27-39 Heart A、2、3...10、J、Q、K
** 40-52 Spade A、2、3...10、J、Q、K
*/
void getPokerDesc(int num){
    const char *poker_desc[] = {"A","2","3","4","5","6","7","8",
                                "9","10","Jack","Queen","King"};

    if (num>0 && num<14)
        cout<<" Club "<<poker_desc[num-1];
    else if (num>=14 && num<27)
        cout<<" Diamond "<<poker_desc[num-14];
    else if (num>=27 && num<40)
        cout<<" Heart "<<poker_desc[num-27];
    else if (num>=40 && num<=52)
        cout<<" Spade "<<poker_desc[num-40];
    else
        cout<<"ERROR!!!";
}

/*
** Return card points
** 1-13 Club A、2、3...10、J、Q、K
** 14-26 Diamond A、2、3...10、J、Q、K
** 27-39 Heart A、2、3...10、J、Q、K
** 40-52 Spade A、2、3...10、J、Q、K

```

```

*/
int getPoint(int num){
    if (num>0 && num<14) {
        if (num>10)
            return 10;
        else
            return num;
    }
    else if (num>=14 && num<27) {
        if (num>23)
            return 10;
        else
            return num-13;
    } else if (num>=27 && num<40) {
        if (num>36)
            return 10;
        else
            return num-26;
    } else if (num>=40 && num<=52) {
        if (num>49)
            return 10;
        else
            return num-39;
    } else {
        return 0;
    }
}

/* Input data to a file for the game's record
*
*/

void savePlayerScore(int score){
    ofstream outfile;

    outfile.open("record.dat", ios::app);
    if(!outfile) {
        cout<<"File Open ERROR."<<endl;
    } else {
        outfile << score << endl;
        outfile.close();
    }
}

```



```

    }
}

//Read data of final scores from a file
int getPlayerScore(){
    ifstream infile;
    string buf;
    int score;

    infile.open("record.dat", ios::in);
    if(!infile) return 0;
    score = 0;
    while(getline(infile,buf)) {
        score += atoi(buf.c_str());
    }

    infile.close();

    return score;
}

//Get Winning Rate by calculating all of data
double getWinRate(){
    ifstream infile;
    string buf;
    int play, win;

    infile.open("record.dat", ios::in);
    if(!infile) return 0;

    play = 0;
    win = 0;
    while(getline(infile,buf)) {
        play++;
        if (atoi(buf.c_str()) > 0) win++;
    }
}

```

```
infile.close();
```

```
if (play > 0 ) {  
    return 1.0 * win / play*100;  
} else {  
    return 0;  
}  
}
```