Project 2

<Black Jack>

CIS-5

Name: Junlong Wu

Date: 06/02/17

Introduction

Title: Black Jack

Blackjack, also known as twenty-one, is the most widely played casino banking game in the world. Blackjack is a comparing card game between a player and dealer, meaning players compete against the dealer but not against other players. It is played with one or more decks of 52 cards. The objective of the game is to beat the dealer in one of the following ways:

- Get 21 points on the player's first two cards (called a "blackjack" or “natural"), without a dealer blackjack;
- Reach a final score higher than the dealer without exceeding 21; or
- Let the dealer draw additional cards until his or her hand exceeds 21.

Summary

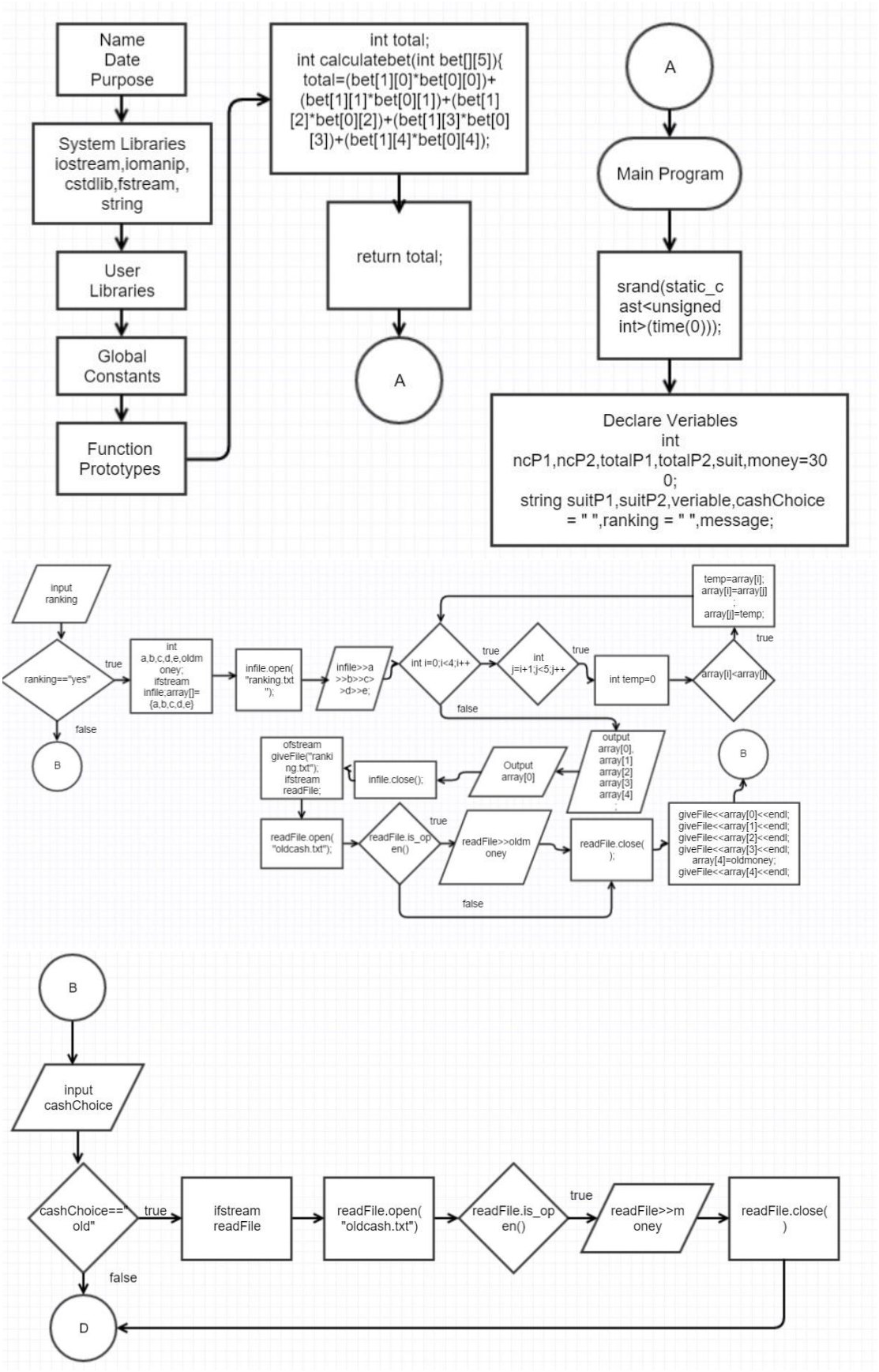
Project size: 257 lines

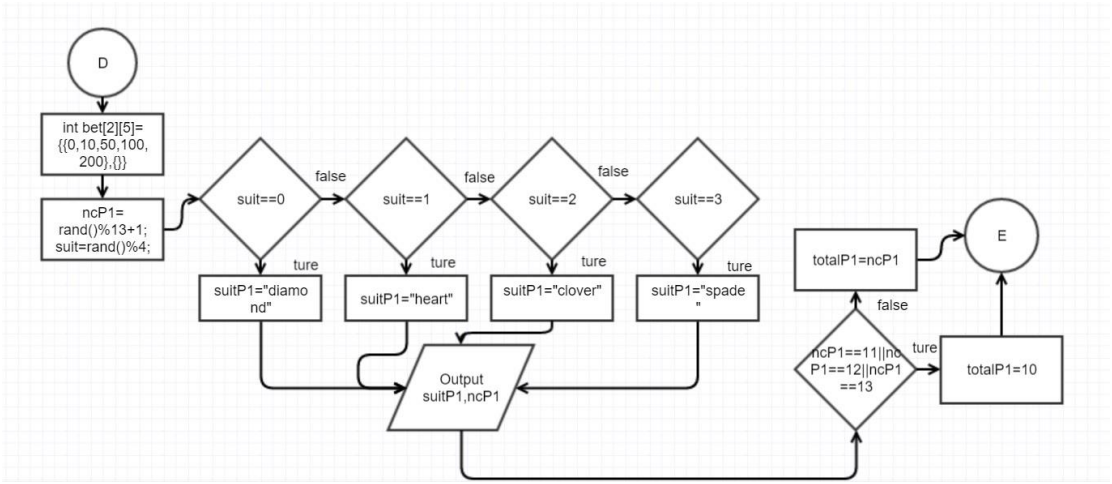
The number of variables: 23

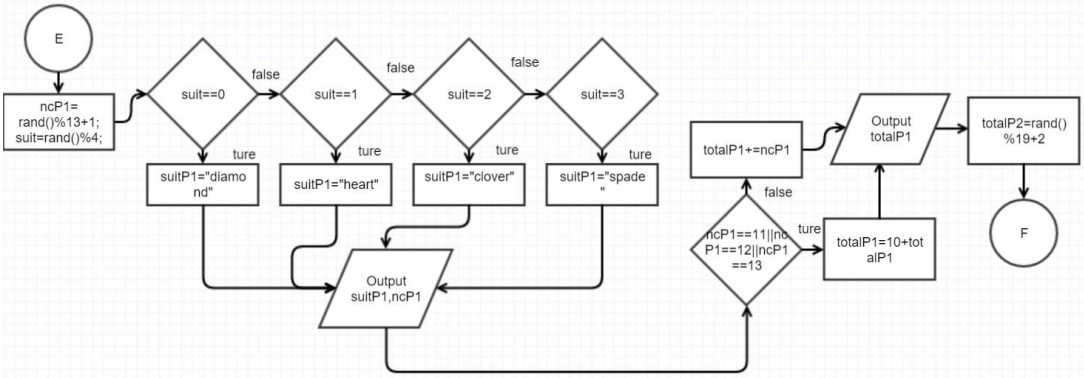
It is my second time to do the project. This time I think I did better than last time. I add more systems in my game, and text it again and again to make it perfect. I am very satisfied with this program. I wish I can make a better one in the future.

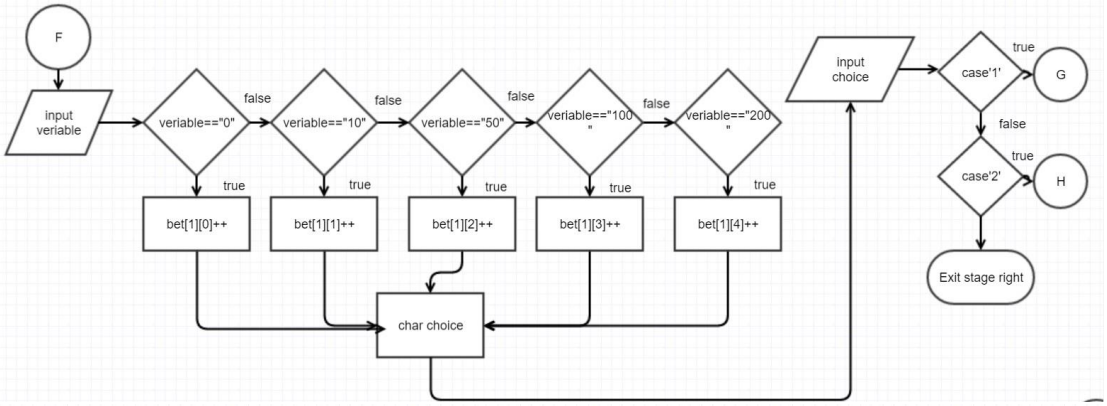
FlowChart Pseudo Code

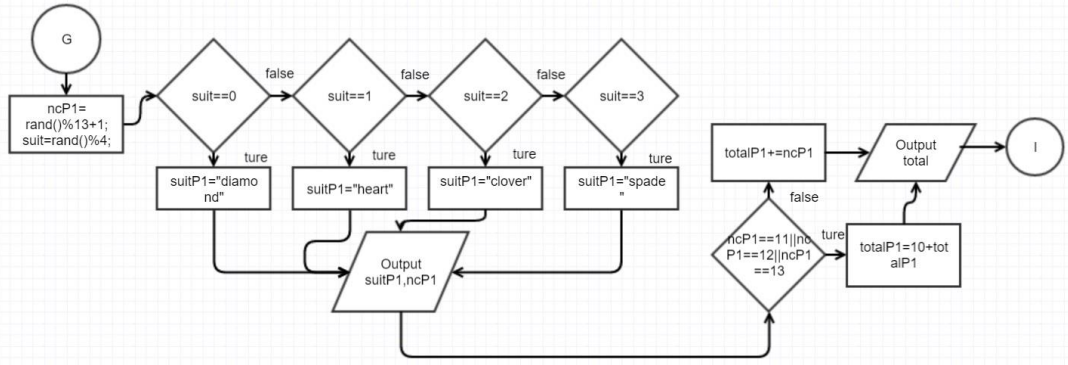
Since my flowchart is really long, I break it to nine small pieces.

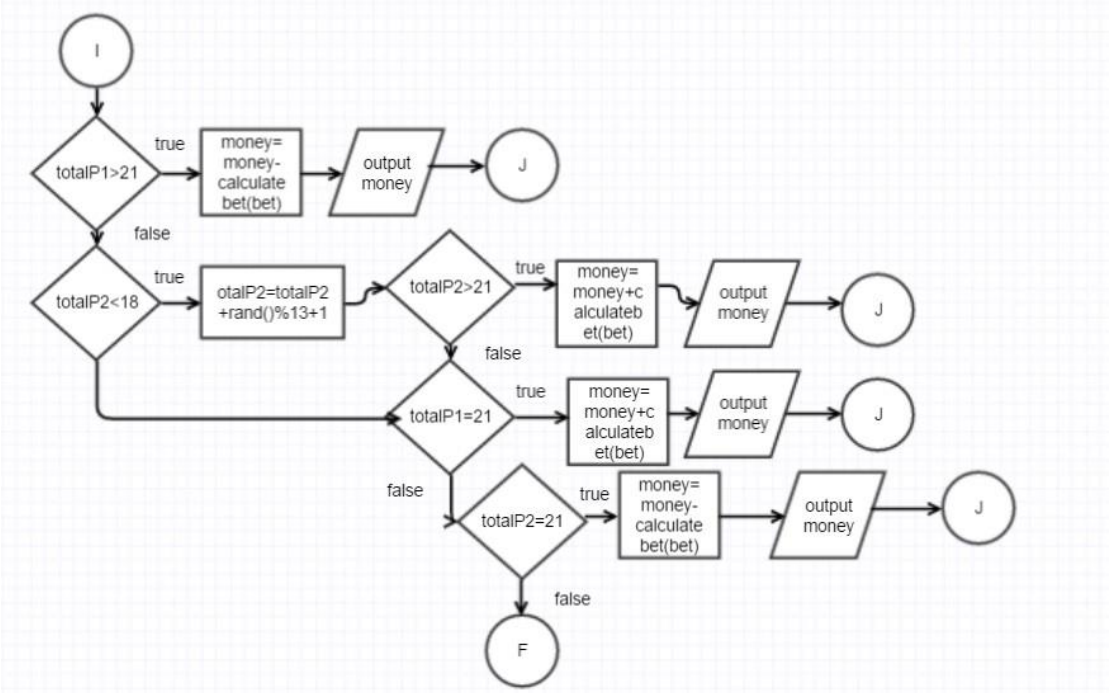

Name
Date
Purpose
System Libraries
iostream,iomanip,
cstdlib,fstream,
string
User
Libraries
Global
Constants
Function
Prototypes
int total;
int calculatebet(int bet[][5]){
total=(bet[1][0]*bet[0][0])+
(bet[1][1]*bet[0][1])+(bet[1]
[2]*bet[0][2])+(bet[1][3]*bet[0]
[3])+(bet[1][4]*bet[0][4]);
return total;
A
A
Main Program
srand(static_c
ast<unsigned
int>(time(0)));
Declare Veriables
int
ncP1,ncP2,totalP1,totalP2,suit,money=30
0;
string suitP1,suitP2,veriable,cashChoice
= " ",ranking = " ",message;
input
ranking
ranking=="yes"
true
false
B
int
a,b,c,d,e,oldm
oney;
ifstream
infile;array[]=
{a,b,c,d,e}
infile.open(
"ranking.txt
");
infile>>a
>>b>>c
>>d>>e;
int i=0;i<4;i++
true
int
j=i+1;j<5;j++
true
int temp=0
temp=array[i];
array[i]=array[j]
;
array[j]=temp;
true
array[i]<array[j]
false
ofstream
giveFile("ranki
ng.txt");
ifstream
readFile;
infile.close();
Output
array[0]
output
array[0],
array[1]
array[2]
array[3]
array[4]
;
B
readFile.open(
"oldcash.txt");
readFile.is_op
en()
true
readFile>>oldm
oney
readFile.close(
);
giveFile<<array[0]<<endl;
giveFile<<array[1]<<endl;
giveFile<<array[2]<<endl;
giveFile<<array[3]<<endl;
array[4]=oldmoney;
giveFile<<array[4]<<endl;
false
B
input
cashChoice
cashChoice=="
old"
true
false
D
ifstream
readFile
readFile.open(
"oldcash.txt")
readFile.is_op
en()
true
readFile>>m
oney
readFile.close(
)

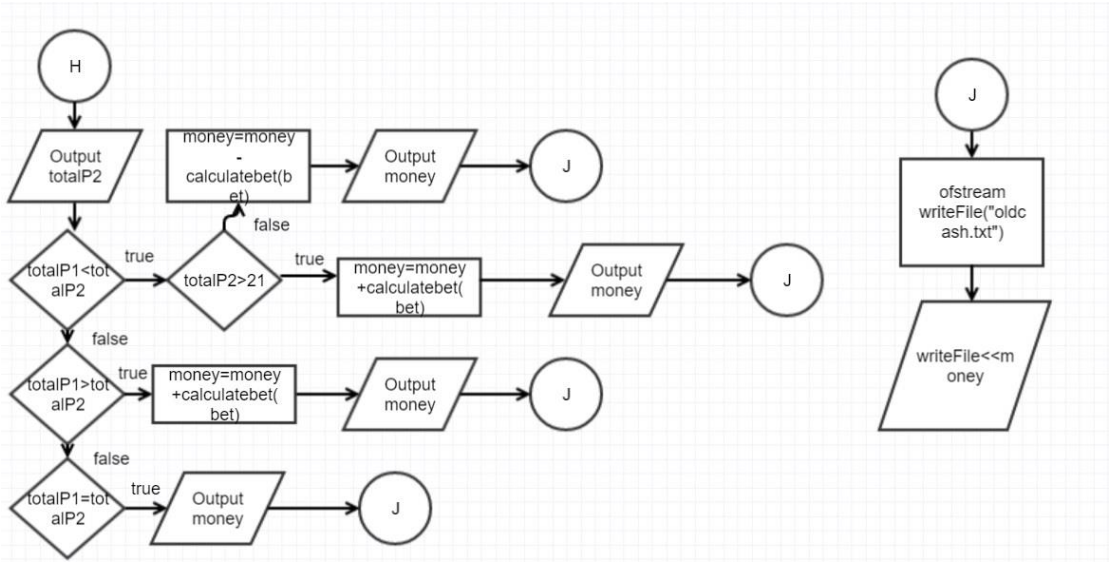

D
int bet[2][5]={{0,10,50,100,200},{}}
ncP1=rand()%13+1;suit=rand()%4;
suit==0
suit==1
suit==2
suit==3
false
ture
suitP1="diamond"
suitP1="heart"
suitP1="clover"
suitP1="spade"
Output suitP1,ncP1
totalP1=ncP1
E
ncP1==11||ncP1==12||ncP1==13
totalP1=10


E
ncP1=rand()%13+1;suit=rand()%4;
suit==0
suit==1
suit==2
suit==3
false
ture
suitP1="diamond"
suitP1="heart"
suitP1="clover"
suitP1="spade"
Output suitP1,ncP1
totalP1+=ncP1
Output totalP1
totalP2=rand()%19+2
F
ncP1==11||ncP1==12||ncP1==13
totalP1=10+totalP1


F
input veriable
veriable=="0"
veriable=="10"
veriable=="50"
veriable=="100"
veriable=="200"
false
true
bet[1][0]++
bet[1][1]++
bet[1][2]++
bet[1][3]++
bet[1][4]++
char choice
input choice
case'1'
G
case'2'
H
Exit stage right


G
ncP1=rand()%13+1;suit=rand()%4;
suit==0
suit==1
suit==2
suit==3
false
ture
suitP1="diamond"
suitP1="heart"
suitP1="clover"
suitP1="spade"
Output suitP1,ncP1
totalP1+=ncP1
Output total
I
ncP1==11||ncP1==12||ncP1==13
totalP1=10+totalP1





Constructs & Concepts Utilized

iostream Library

Name	Frequency	Description	Location
static_cast	1	Statically cast as different variable	Line 28
cout	36	Output Data	Throughout
cin	4	Input Data	Line 33,74,134,152

cstdlib Library

Name	Frequency	Description	Location
srand()	1	Random # seed	Line 28

rand()	8	Generates rand #	Line 87,88,108,109,130, 157,158,189

string Library

Name	Frequency	Description	Location
string	1	Declare var.	Line 30

fstream Library

Name	Frequency	Description	Location
readFile.open()	2	Open file	Line 59,77
readFile.close()	2	Close file	Line 63,81
inFIle.open()	1	Open file	Line 37
inFile.close()	1	Close file	Line 56
writeFile	1	Write to file	Line 247
giveFile	6	Write to file	Line 64,65,66,67,68, 69
ofstream	2	Declare var.	Line 57,246
ifstream	3	Declare var.	Line 36,58,76

Data Types:

Data Types	Frequency	Location
int	11	Line 21,29,40,41,42,85
unsigned int	1	Line 28
char	1	Line 150
string	6	Line 30
ofstream	2	Line 57,246
ifstream	2	Line 36,58,76

Conditional Statements:

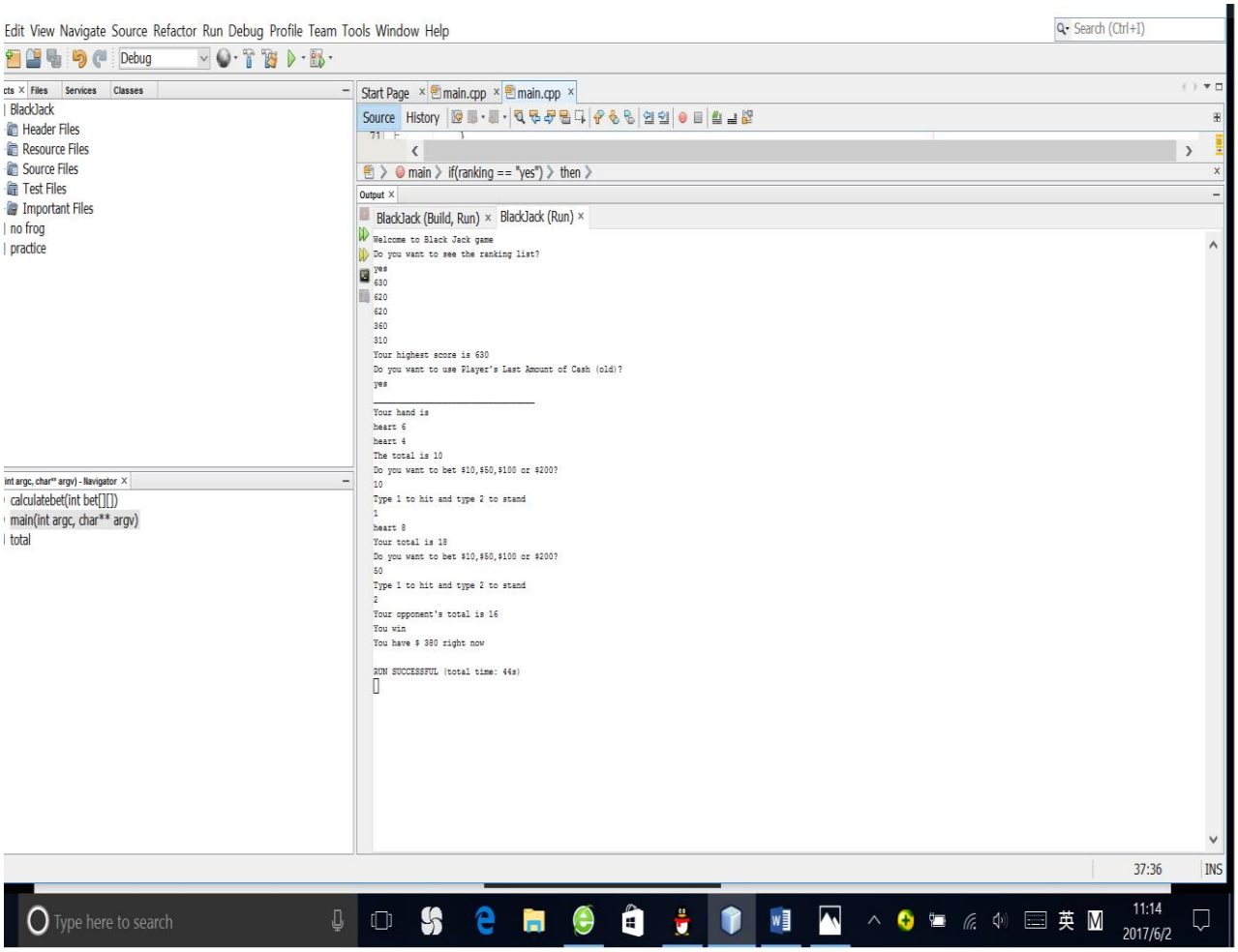
Conditional Statement	Frequency	Location
if	15	Line 34,43,60,75,135,138, 141,144,147,179,188,190, 199,205,217
if/else	3	Line 102,123,172
if/else if	4	Line 89,110,159,216
switch	1	Line 155

Loops:

Loops	Frequency	Location
for	2	Line 40,41
do-while	1	Line 132

Proof of a Working Product

I have a screenshot to prove my game works.



Program

```
//System Libraries
#include <iostream>
#include <iomanip>
#include <cstdlib>
#include <fstream>
#include <string>
using namespace std; //Name-space under which system libraries exist

//User Libraries

//Global Constants

//Function Prototypes
int total;
```

```
int calculatebet(int bet[][5]){
    total=(bet[1][0]*bet[0][0])+(bet[1][1]*bet[0][1])+(bet[1][2]*bet[0][2])+(bet[1][3]*bet[0][3])+(bet[1][4]*bet[0][4]);
return total;
}

int main(int argc, char** argv) {
    srand(static_cast<unsigned int>(time(0)));
    int ncP1,ncP2,totalP1,totalP2,suit,money=300;
    string suitP1,suitP2,veriable,cashChoice = " ",ranking = " ",message;
    cout<<"Welcome to Black Jack game"<<endl;
    cout<<"Do you want to see the ranking list?"<<endl;
    cin>>ranking;
    if(ranking == "yes"){
        int a,b,c,d,e,oldmoney;
        ifstream infile;
        infile.open("ranking.txt");
        infile>>a>>b>>c>>d>>e;
        int array[]={a,b,c,d,e};
        for(int i=0;i<4;i++){
            for(int j=i+1;j<5;j++){
                int temp=0;
                if (array[i]<array[j]){
                    temp=array[i];
                    array[i]=array[j];
                    array[j]=temp;
                }
            }
        }
        cout<<array[0]<<endl;
        cout<<array[1]<<endl;
        cout<<array[2]<<endl;
        cout<<array[3]<<endl;
        cout<<array[4]<<endl;
        cout<<"Your highest score is "<<array[0]<<endl;
            infile.close();
```

```
    ofstream giveFile("ranking.txt");
    ifstream readFile;
    readFile.open("oldcash.txt");
    if(readFile.is_open()){
        readFile>>oldmoney;
    }
    readFile.close();
    giveFile<<array[0]<<endl;
    giveFile<<array[1]<<endl;
    giveFile<<array[2]<<endl;
    giveFile<<array[3]<<endl;
    array[4]=oldmoney;
    giveFile<<array[4]<<endl;

    }

cout<<"Do you want to use Player's Last Amount of Cash (old)?"<<endl;;
cin>>cashChoice;
if(cashChoice=="yes"){
    ifstream readFile;
    readFile.open("oldcash.txt");
    if(readFile.is_open()){
        readFile>>money;
    }
    readFile.close();

}
 cout<<"_____"<<endl;
int bet[2][5]={{0,10,50,100,200},{}};
cout<<"Your hand is"<<endl;
ncP1= rand()%13+1;
suit=rand()%4;
if (suit==0){
    suitP1="diamond";
}
```

```
  else if (suit==1){
     suitP1="heart";
  }
  else if (suit==2){
     suitP1="clover";
  }
  else if (suit==3){
     suitP1="spade";
  }
  cout<<suitP1<<" "<<ncP1<<endl;
  if (ncP1==11||ncP1==12||ncP1==13){
     totalP1=10;
  }
  else {
     totalP1=ncP1;
  }
  ncP1= rand()%13+1;
  suit=rand()%4;
  if (suit==0){
     suitP1="diamond";
  }
  else if (suit==1){
     suitP1="heart";
  }
  else if (suit==2){
     suitP1="clover";
  }
  else if (suit==3){
     suitP1="spade";
  }
  cout<<suitP1<<" "<<ncP1<<endl;
if (ncP1==11||ncP1==12||ncP1==13){
     totalP1=10+totalP1;
  }
  else {
```

```
    totalP1+=ncP1;
}
cout<<"The total is "<<totalP1<<endl;
totalP2=rand()%19+2;

do{
    cout<<"Do you want to bet $10,$50,$100 or $200?"<<endl;
    cin>>veriable;
    if(veriable=="0"){
    bet[1][0]++;
   }
    if(veriable=="10"){
    bet[1][1]++;
   }
   if(veriable=="50"){
    bet[1][2]++;
   }
   if(veriable=="100"){
    bet[1][3]++;
   }
   if(veriable=="200"){
    bet[1][4]++;
   }
    char choice;
    cout<<"Type 1 to hit and type 2 to stand"<<endl;
    cin>>choice;


    switch(choice){
        case '1':
            ncP1= rand()%13+1;
            suit=rand()%4;
            if (suit==0){
                suitP1="diamond";
            }
```

```
else if (suit==1){
   suitP1="heart";
}
else if (suit==2){
   suitP1="clover";
}
else if (suit==3){
   suitP1="spade";
}
cout<<suitP1<<" "<<ncP1<<endl;
if (ncP1==11||ncP1==12||ncP1==13){
   totalP1=10+totalP1;
}
else {
   totalP1+=ncP1;
}
cout<<"Your total is "<<totalP1<<endl;
if (totalP1>21) {
    cout<<"You bust ,so you lose"<<endl;
    money=money-calculatebet(bet);
    cout<<"You have $ "<<money<<" right now"<<endl;
    goto leave_loop;

    }


if (totalP2<18) {
    totalP2=totalP2+rand()%13+1;
        if (totalP2>21){
        cout<<"Your opponent bust, and you win "<<endl;
        cout<<"Your opponent's total is "<<totalP2<<endl;
        money=money+calculatebet(bet);
        cout<<"You have $ "<<money<<" right now"<<endl;
        goto leave_loop;
```

```
            }
        }
        if (totalP1==21) {
            cout<<"You win "<<endl;
            money=money+calculatebet(bet);
            cout<<"You have $ "<<money<<" right now"<<endl;
            goto leave_loop;
        }
        if (totalP2==21) {
            cout<<"Your opponent win "<<endl;
            money=money-calculatebet(bet);
            cout<<"You have $ "<<money<<" right now"<<endl;
            goto leave_loop;
        }



        break;
    case '2':
        cout<<"Your opponent's total is "<<totalP2<<endl;
        if (totalP1<totalP2) {
            if (totalP2>21){
                cout<<"Your opponent bust, and you win "<<endl;
                money=money+calculatebet(bet);
                cout<<"You have $ "<<money<<" right now"<<endl;
                goto leave_loop;
            }
            cout<<"You lose "<<endl;
            money=money-calculatebet(bet);
            cout<<"You have $ "<<money<<" right now"<<endl;
            goto leave_loop;
        }
        else if (totalP1>totalP2) {
            cout<<"You win "<<endl;
            money=money+calculatebet(bet);
            cout<<"You have $ "<<money<<" right now"<<endl;
```

```
                goto leave_loop;
            }
            else if (totalP1==totalP2) {
                cout<<"You tie "<<endl;
                cout<<"You have $ "<<money<<" right now"<<endl;
                goto leave_loop;
            }
            break;
        default:
            cout<<"You press the wrong number "<<endl;
            return 0;
        }
    }while(1);
    leave_loop: ;
    ofstream writeFile("oldcash.txt");
    writeFile<<money;




    //Exit stage right!
    return 0;
}
```