Project 1

Title:

21 the card game

Course:

Due Date:

July 21, 2015

Author:

Hector Espinoza

Title:

21 the card game

Introduction:

21 is played with an international set of 52 cards. Each player is dealt two cards on the first round. When the game begins the player looks at their cards and o hold two or more cards which together sum to 21 (ideally an Ace and a tenth, this is known as pontoon or a natural) but above 15 and as near to 21 as possible. Any hand that exceeds 21 is "bust" and the player loses that round.

How to play:

Each player has an independent game against the dealer and the objective is to beat the dealer. If the player's cards total is higher than 21, it is called a "bust" and the dealer wins, even if the dealer have a bust as well and if the dealer and the player have the same value it's a "bust". A 21 hand value is calculated as follows: Cards of 2 - 10 are worth their face value. Jack, Queen and King are also worth 10 each. The Ace card is worth 11, unless it causes the player to bust his hand and in this case the Ace card can be counted as 1. A 2 cards hand with ace and a 10 value card is called a "blackjack" and it is a winning hand unless the dealer have an ace and a 10 value 2 cards hand is well, in this case it is a "bust".

The game course:

After all players have placed their bets, the dealer deals each player with 2 cards including himself, the dealers first card will be face up and the other is face down.

21 player play options:

1. Hit - Get another card.

2. Stand - Stay with the current hand..

After all players have played their hand, the dealer revels his other card and start to play his hand.
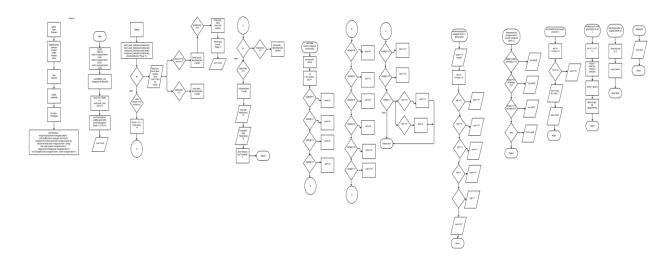
House rules say that the dealer must continue to hit until he has at least 17.

Summary:

Project Size: 150+ lines

I tried to make this game simple and easy to use, my main vision for this game was to be based off of simplicity so that the player will have fun playing and can play for perhaps hours. I think that I made it work really well, I so far am addicted to it and want to keep playing it over and over. A way that I thought would be best so that the game would be fast and fun was to be player versus computer. I believe that I worked it out really well and made the game a fair game so that the computer and player felt that it wouldn't be one sided. Another thing I tried to implement was the use of only 4 of each number, I could have set a random number generator and get any number a multiple of times. Just in a real life game you cannot pull a card more than 4 times and I was fortunate enough to use an array and be able to put that restriction on the game.

Flow chart:

Pseudo code:

Random generator, generates numbers from 1 to 13 to get all the cards of the 52 deck (without jokers).

Calls the Print card function.

For loops to the size and does them all from 1 to 13 and at each row it will create a new row once it reaches 13.

Calls the shuffle function.

Uses a for loop and random number generator then swaps the positions in the array.

Calls the print card function.

For loops to the size and does them all from 1 to 13 and at each row it will create a new row once it reaches 13.

Uses a for loop to cout 25 endls to clear the board.

Calls the show hand function.

Uses a for loop to count all the items in the array. Couts the name of the card holder. Then uses if statements to determine if it is a 13 it equals a K. If it is a 12 it equals a Q. if it is an 11 it equals a J. If it is a 1 it equals an A. Then every other number couts as its original face value.

Then do whiles to determine to draw another card.

Uses an if statement to determine if the choice is a yes or no. If it is a Yes then draw another card, then will show the users hand. After outputting the cards it ask the user if he wants to draw another card.

While if the answer is anything other than yes it will loop and to the size of the array and will to determine if the computer needs to draw another card or not.

Calls the compare function.

If player is greater than computer and is less than 21 then the player wins. If the player is equal to 21 the player wins. If the player is less than 21 and computer is greater than 21 the player wins. Any other condition the player loses.

The next step couts the players total hand.

Calls the total function for player 1.

Uses for loops from 2 to 10 and sums them up for their face value, jack queen and king have a value of 10 and the Ace has a value of 1 or 11 based on if it passes the max limit of 21.

Then couts the computers total hand.

Uses for loops from 2 to 10 and sums them up for their face value, jack queen and king have a value of 10 and the Ace has a value of 1 or 11 based on if it passes the max limit of 21.

Then finally will display the computers hand.

Calls the show hand function for the computer.

Major Variables

Function	Decscription	location
Cin/cout	Used to input and output information	Throughout the code
Local scope	The sum of the total of cards	In function unsigned short total
Global scope	Const int SIZE	In main
char	Choice to ask the user if he wants to draw another card	In main
short	Used to set the size of the vectors	In main the vectors
string	Computer name and player name	In show hand function
comments	Used to comment the code	Throughout the code
If ,If-else	To determine a condition	In the function total
While,do-while	To determine a condition to be true or false, and will loop till it is true.	In main
for	Is a counter for what in the array	In the show hand function
protoypying	Making functions	Before main
Void vs. return	Used to display rules	After main
Random number generator	Generates random numbers	In print cards function
array	Creates a table of information	In print cards
vector	Used to call the functions in the library	In function shuffle

Pop_back	Push the cards back and deletes the last one	Draw card function
Push_back	Pushed the desired slot to the end	In main
Increment and decrement	Counts up 1 or down 1	In the show hand function in the for loop
Set width	Uses to set the desired spacing	In print cards function
casting	Re type a variable	In main, after declare variables
Arithmetic operators	Used to determine which hand was greater the player or the computer	In the function compare

Refrence:

1. Cplusplus the online resource(used as a reference for vectors)
2. The book Gaddis 11th Edition

Program:

//System Libraries

#include <iostream>

#include <string>

#include <cstdlib>

#include <iomanip>

#include <vector>

using namespace std;

//User Libraries

//Global Constants

//Function Prototypes

void Display();   //Displays the rules of the game

void printCards(vector<unsigned short>);   //Prints cards 4 of each

void shuffle(vector<unsigned short>&,int);   //Shuffles the cards

unsigned short drawCard(vector<unsigned short>&); //Draws one card at a time from the bottom of the deck

void showHand(vector<unsigned short> , string);   //Shows the players hand

void comp1(vector<unsigned short>);   //Computers hand of cards

```cpp
unsigned short total(vector<unsigned short>);              //Total of the players card

void compare(vector<unsigned short> ,vector<unsigned short> );   //Compares the computer and the
users cards to determine the winner


//Execution Begins Here!
int main(int argc, char** argv) {

    //Declare Variables

    char choice;                         //Choice to draw another card or not

    const int SIZE=52;                   //Size of the array

    vector<unsigned short> cards;              //Vector for the cards

    vector<unsigned short> hand1;

    vector<unsigned short> comp;

    //Vector for the players hand to store what cards he has

    srand(static_cast<unsigned int>(time(0)));

    for(int i=0; i<SIZE; i++){

        cards.push_back (i%13+1);

    }

    printCards(cards);

    shuffle(cards,SIZE);

    printCards(cards);

    for(int l=1;l<25;l++){

        cout<<endl;

    }

    Display();

    //Draws the players cards

    hand1.push_back(drawCard(cards));

    hand1.push_back(drawCard(cards));

    comp.push_back(drawCard(cards));

    comp.push_back(drawCard(cards));
```

```cpp
//    showHand(comp,"Computer");
   showHand(hand1,"Player 1");
      do{
      cout<<"Would you like to draw another card?(Yes='Y,No='N')"<<endl;
      cin>>choice;
      if (choice=='Y'||choice=='y'){
         hand1.push_back(drawCard(cards));
         if(total(comp)<17){
            comp.push_back(drawCard(cards));
         }
         showHand(hand1,"Player 1");
      }
      cout<<endl;
   }while(choice=='Y'||choice=='y');
   for(int i = 0; i < comp.size(); ++i)
   do{
      if(total(comp)<17){
         comp.push_back(drawCard(cards));
      }
   }while(total(comp)<17);
   compare(hand1,comp);
   cout<<"Your total "<<total(hand1)<<"\n";
   cout<<"Computer's total "<<total(comp)<<"\n";
   showHand(comp,"Computer");
   //Exit stage right!
   return 0;
}
unsigned short total(vector<unsigned short>cards){
   unsigned short sum=0;
```

```cpp
unsigned int ace;

for(int i=0;i<cards.size();i++){

   if(cards[i]==2){

      sum+=2;

   }else if(cards[i]==3){

      sum+=3;

   }else if(cards[i]==4){

      sum+=4;

   }else if(cards[i]==5){

      sum+=5;

   }else if(cards[i]==6){

      sum+=6;

   }else if(cards[i]==7){

      sum+=7;

   }else if(cards[i]==8){

      sum+=8;

   }else if(cards[i]==9){

      sum+=9;

   }else if(cards[i]==10){

      sum+=10;

   }else if(cards[i]==11){

      sum+=10;

   }else if(cards[i]==12){

      sum+=10;

   }else if(cards[i]==13){

      sum+=10;

   }else if(cards[i]==1){

      if(sum<10){

         sum+=11;
```

```cpp
        }
        if(sum>21){
            sum+=1;
        }
    }
}
return sum;
}
void showHand(vector<unsigned short> v, string name){
    cout<<name<<"'s hand\n";
    for(int i=0; i<v.size(); i++){
        if(v[i]==13){
            cout<<"K ";
        }else if(v[i]==12){
            cout<<"Q ";
        }else if(v[i]==11){
            cout<<"J ";
        }else if(v[i]==1){
            cout<<"A ";
        }else cout<<v[i]<<" ";
    }
    cout<<"\n";
}
 void compare(vector<unsigned short> p,vector<unsigned short> c){
    if(total(p)>total(c) && total(p)<=21){
        cout<<"You Win!!!!"<<endl;
    }
    else if(total(p)==21 && total(p)<=21){
         cout<<"You Win!!!"<<endl;
```

```cpp
        }
        else if(total(p)<=21 && total(c)>21){
            cout<<"You win!!!"<<endl;
        }
        else{
            cout<<"You Lose!!!"<<endl;
        }


    }
    void printCards(vector<unsigned short> v){
        for(int i=0; i<v.size(); i++){
            if(i%13==0)cout<<"\n";
            cout<<setw(2)<<v[i]<<" ";
        }
        cout<<endl;
    }
    void shuffle(vector<unsigned short> &v,int s){
        for(int i=0 ; i< s*3 ; i++){
            int pos1 = rand()%s;
            int pos2 = rand()%s;
            unsigned short temp = v[pos1];
            v[pos1]=v[pos2];
            v[pos2]=temp;
        }
    }
    unsigned short drawCard(vector<unsigned short> &c){
        unsigned short temp=c[c.size()-1];
        // deal the fist card, temp = c[0]
        // swap the first and last card, swap(c[0],c[c.size()-1)
```

```cpp
    //c.pop_back()

    c.pop_back();

    return temp;

}

void Display(){

    cout<<"Welcome to the game of 21, based on the popular card game."<<endl;

    cout<<"How to play: The objective of the game is to get the highest total without exceeding
21."<<endl;

    cout<<"Each player has an independent game against the dealer and the objective is to beat the
dealer."<<endl;

    cout<<"If the players hand total is higher then 21, it is called a bust, and the dealer wins."<<endl;

    cout<<"Even if both the player and the dealer have a bust as well."<<endl;

    cout<<"A 21 hand is calculated as follows: Cards of 2-10 are worth their face value."<<endl;

    cout<<"Jack, Queen, King are also worth 10 each."<<endl;

    cout<<"The Ace card is worth 11 unless it causes the player to bust his hand, which in this case the
Ace can be counted as a 1."<<endl;

    cout<<"A two cards hand with the ace and a 10 value is called blackjack and it is a winning
hand,"<<endl;

    cout<<"unless the dealer has an Ace and a 10 value, in this case is a bust."<<endl;

    cout<<endl;

}
```