**INTRODUCTION**

**TO**

**PROGRAMMING**

**INNOVATION**

**PROJECT**

**THE GAME**

**Blackjack**, also known as **twenty-one**, is a comparing [card game](https://en.wikipedia.org/wiki/Card_game) between usually several players and a dealer, where each player in turn competes against the dealer, but players do not play against each other. It is played with one or more [decks](https://en.wikipedia.org/wiki/Playing_card) of 52 cards, and is the most widely played [casino](https://en.wikipedia.org/wiki/Casino_game) banking game in the world.[[1]](https://en.wikipedia.org/wiki/Blackjack#cite_note-1) 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](https://en.wikipedia.org/wiki/Natural_(gambling))"), without a dealer blackjack;
* Reach a final score higher than the dealer without exceeding 21; or
* Let the dealer draw additional cards until their hand exceeds 21.

Players are each dealt two cards, face up or down depending on the casino and the table at which you sit. In the U.S., the dealer is also dealt two cards, normally one up (exposed) and one down (hidden). In most other countries, the dealer receives one card face up. The value of cards two through ten is their pip value (2 through 10). Face cards (Jack, Queen, and King) are all worth ten. Aces can be worth one or eleven. A hand's value is the sum of the card values. Players are allowed to draw additional cards to improve their hands. A hand with an ace valued as 11 is called "soft", meaning that the hand will not bust by taking an additional card; the value of the ace will become one to prevent the hand from exceeding 21. Otherwise, the hand is "hard"

HEADER FILES

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

FUNCTIONS DEFINED

void play(); start the game

void turn(); execute turn of player and dealer(computer)

int hit(int \*val); drawing cards again

char \*cface(int v); return array of face of card

char \*csuit(int v); return array of suit of card

int cval(int v); return value of the card

void bet(); betting some value from balance

void result(); result of playing game

void initdeck(); deck for the player

void shuffle(); randomly assigning values to player deck

void checkAce(int v1, int v2, int \*a1); checking the value of ace to be taken

void quit(); quit the game

SOURCE CODE

**#include <stdio.h>**

**#include <stdlib.h>**

**#include <string.h>**

**#define CARDS 52**

**int balance;**

**int pot;**

**int deck[52]; //every card has a number assigned in this array**

**int sdeck[52]; //player's start deck**

**void play();**

**void turn();**

**int hit(int \*val);**

**char \*cface(int v);**

**char \*csuit(int v);**

**int cval(int v);**

**void bet();**

**void dealer();**

**void result();**

**void initdeck();**

**void shuffle();**

**void checkAce(int v1, int v2, int \*a1);**

**void quit();**

**int main(){**

**printf("Welcome to my BlackJack Simulator! Below are the basic rules:\n- Beat the dealer's hand without going over 21.\n- Face cards are worth 10, Aces are worth 1 or 11, whichever makes a better hand.\n- Each player starts with two cards, and one of the dealer's cards is hidden until the end.\n- Type 'hit' to ask for another card. Type 'stand' to hold your total and end your turn.\n- If you go over 21 you bust, and the dealer wins regardless of his hand.\n- If you are dealt 21 from the start (Ace & 10), you got a blackjack. If you get a blackjack, you win 1.5 times the amount of your bet automatically, unless the dealer also gets a blackjack, in which case it is a push.\n- Remember: Type 'hit' to get another card, and 'stand' to hold. At the beginning of the round, type 'bet' followed by the quantity you want to bet (i.e. 'bet 50').\nType 'play' to begin. At any time, you may type 'help' to get a list of valid commands.\n");**

**char input[6];**

**scanf("%5s", input);**

**while(strcmp(input, "quit") != 0)**

**{**

**while(strcmp(input, "play") != 0)**

**{**

**if(strcmp(input, "help") == 0)**

**{**

**printf("Type 'play' to begin the game.\n");**

**}**

**else if(strcmp(input, "quit")==0)**

**{**

**quit();**

**printf("Enter 'play' to start the game.\n");**

**}**

**else**

**{**

**printf("Invalid command, type 'play' to begin.\n");**

**scanf("%s", input);**

**}**

**scanf("%5s", input);**

**}**

**play();**

**scanf("%5s", input);**

**}**

**}**

**void play()**

**{**

**printf("Lets start you off with 500 credits.\n");**

**balance = 500;**

**while(balance>0)**

**{**

**bet();**

**initdeck();**

**shuffle();**

**turn();**

**}**

**printf("Sorry, it looks like you lost! Type 'play' to play another game.\n");**

**}**

**void turn(){**

**int inc=0;**

**int d1 = hit(&inc);**

**char \*cn1 = csuit(d1); // this returns card face array base address to pointer**

**char \*cf1 = cface(d1); // this returns card face array base address to pointer**

**int cv1 = cval(d1);**

**printf("You got a %s of %s worth %d, and ", cf1, cn1, cv1);**

**int d2 = hit(&inc); //second drawing of card**

**char \*cn2 = csuit(d2);**

**char \*cf2 = cface(d2);**

**int cv2 = cval(d2);**

**printf("a %s of %s worth %d.\n", cf2, cn2, cv2);**

**int d3 = hit(&inc); //third drawing of card**

**char \*cn3 = csuit(d3);**

**char \*cf3 = cface(d3);**

**int cv3 = cval(d3);**

**printf("The dealer's face up card is a %s of %s worth %d.\n", cf3, cn3, cv3);**

**int d4 = hit(&inc); //fourth drawing of card**

**char \*cn4 = csuit(d4);**

**char \*cf4 = cface(d4);**

**int cv4 = cval(d4);**

**int ptotal = cv1 + cv2; // player total value**

**int dtotal = cv3; // dealer total value**

**checkAce(cv1, cv2, &ptotal); // deciding if ace value 1 or 11 is to be taken**

**printf("You have a total of %d points, and the dealer has %d.\n", ptotal, dtotal);**

**if(ptotal == 21)**

**{**

**printf("Congrats! You got a blackjack!! Your payout is 2-to-1, %d.\n", (int)(pot\*2)); //double win if player total becomes 21**

**balance += (pot\*2);**

**}**

**else**

**{**

**char input[6];**

**scanf("%5s", input); //asking player if he wants to stay on current value or draw more cards**

**while(strcmp(input,"stand")!=0)**

**{**

**if(strcmp(input,"hit") == 0)**

**{**

**int val = hit(&inc);**

**char \*ns = csuit(val); //new suit**

**char \*nf = cface(val); //new face**

**int nv = cval(val); //new value**

**ptotal += nv;**

**printf("You got a %s of %s worth %d.\n", nf, ns, nv);**

**if(ptotal < 21)**

**{**

**if(nv==1)**

**{**

**if(ptotal+10 == 21)**

**{**

**ptotal += 10;**

**printf("Congrats! You got 21! Payout is 2-to-1.\n");**

**balance += (pot\*2);**

**break;**

**}**

**else**

**{**

**int input;**

**printf("You've got an ace. Choose whether you want to make it count as 1 or 11.\n");**

**scanf("%d",&input);**

**if(input == 11)**

**ptotal += 10;**

**}**

**}**

**}**

**else if(ptotal == 21)**

**{**

**printf("Congrats! You got 21!\n");**

**break;**

**}**

**else**

**{**

**printf("Oh no, you've busted with %d. Try Again!\n", ptotal);**

**break;**

**}**

**}**

**else if(strcmp(input,"help")==0)**

**{**

**printf("Type 'hit' to be dealt another card. Type 'stand' to hold.\n");**

**}**

**else if(strcmp(input, "quit")==0)**

**{**

**quit();**

**printf("Enter 'hit' or 'stand'.\n");**

**}**

**else**

**{**

**printf("Invalid command, try again.\n");**

**printf("Your new total is %d.\n", ptotal);**

**scanf("%5s", input);**

**}**

**}**

**}**

**if(ptotal < 21)**

**{**

**printf("The dealer's flips a %s of %s worth %d.\n", cf4, cn4, cv4);**

**dtotal += cv4;**

**if(cv4 == 1)**

**{**

**if(dtotal+11 < 21)**

**dtotal += 10;**

**}**

**if(dtotal >= 16)**

**{**

**printf("The dealer stands with %d.\n", dtotal);**

**}**

**while(dtotal < 16)**

**{**

**int val = hit(&inc);**

**char \*ns = csuit(val);**

**char \*nf = cface(val);**

**int nv = cval(val);**

**dtotal += nv;**

**printf("The dealer got a %s of %s worth %d.\n", nf, ns, nv);**

**if(dtotal < 16)**

**{**

**if(nv==1)**

**{**

**if(dtotal+11<21)**

**dtotal += 10;**

**}**

**}**

**else if(dtotal == 21)**

**{**

**printf("The dealer just got 21.\n");**

**break;**

**}**

**else if(dtotal > 21)**

**{**

**printf("The dealer busted with %d! You win!\n", dtotal);**

**break;**

**}**

**else{**

**printf("The dealer stands with %d.\n", dtotal);**

**break;**

**}**

**printf("The dealer's new total is %d.\n", dtotal);**

**}**

**if(dtotal<ptotal)**

**{**

**printf("You beat the dealer! Your payout is %d.\n", (int)(pot\*1.5));**

**balance += (pot\*1.5);**

**}**

**else if(dtotal==ptotal)**

**{**

**printf("Its a tie! Push pot, 1-to-1 payout of %d.\n", pot);**

**balance += pot;**

**}**

**else if(dtotal>ptotal && dtotal <= 21)**

**{**

**printf("Oh no! Looks like the dealer won. Try again!\n");**

**}**

**else**

**{**

**printf("You beat the dealer! Your payout is %d.\n", (int)(pot\*1.5));**

**balance += (pot\*1.5);**

**}**

**}**

**if(balance > 0)**

**printf("Your new balance is %d.\n", balance);**

**}**

**void checkAce(int v1, int v2, int \*a1)**

**{**

**if(v1 == 1 || v2 == 1)**

**{**

**int input;**

**if(v1 == 1 && v2 == 1)**

**{**

**v1 = 11;**

**v2 = 1;**

**\*a1 = 12;**

**printf("Since you got 2 Aces, we set one to be worth 11 and the other to be worth 1.\n");**

**}**

**else if(v1 == 1 || v2 == 1)**

**{**

**if(v1+10+v2 == 21)**

**\*a1 == 21;**

**else**

**{**

**printf("You've got an ace. Choose whether you want to make it count as 1 or 11.\n");**

**scanf("%d",&input);**

**if(input == 11)**

**\*a1 += 10;**

**}**

**}**

**}**

**}**

**int cval(int v)**

**{**

**int a;**

**if(((v%13)+1)<10)**

**a = v%13+1;**

**else**

**a = 10;**

**return a;**

**}**

**char \*csuit(int v)**

**{**

**v = v%4;**

**switch(v)**

**{**

**case 0 :**

**return "Hearts";**

**case 1 :**

**return "Clubs";**

**case 2 :**

**return "Diamonds";**

**case 3 :**

**return "Spades";**

**}**

**}**

**char \*cface(int v)**

**{**

**v = v%13+1;**

**char \*output;**

**switch(v)**

**{**

**case 1 :**

**output = "Ace";**

**break;**

**case 2 :**

**output = "Two";**

**break;**

**case 3 :**

**output = "Three";**

**break;**

**case 4 :**

**output = "Four";**

**break;**

**case 5 :**

**output = "Five";**

**break;**

**case 6 :**

**output = "Six";**

**break;**

**case 7 :**

**output = "Seven";**

**break;**

**case 8 :**

**output = "Eight";**

**break;**

**case 9 :**

**output = "Nine";**

**break;**

**case 10 :**

**output = "Ten";**

**break;**

**case 11 :**

**output = "Jack";**

**break;**

**case 12 :**

**output = "Queen";**

**break;**

**case 13 :**

**output = "King";**

**}**

**return output;**

**}**

**void quit()**

**{**

**printf("Are you sure you want to quit? Type 'y' or 'n'.\n");**

**char input[6];**

**scanf("%s",input);**

**if(strcmp(input,"y")==0)**

**exit(0);**

**else**

**printf("Quit cancelled.\n");**

**}**

**int hit(int \*i)**

**{**

**int a = sdeck[\*i];**

**\*i = \*i + 1;**

**return a; //returning the variable with start deck**

**}**

**void bet(){**

**char input[6];**

**int val;**

**printf("Enter an amount you would like to bet.\n");**

**scanf("%5s", input);**

**while(strcmp(input, "bet") != 0)**

**{**

**if(strcmp(input,"help") == 0)**

**printf("Type 'bet' followed by a bet amount. For example, you could type 'bet 50'.\n");**

**else if(strcmp(input,"quit")==0)**

**{**

**quit();**

**printf("Enter a bet amount.\n");**

**}**

**else**

**printf("You've entered an invalid command. Type 'help' for a list of valid commands.\n");**

**scanf("%5s", input);**

**}**

**scanf("%d", &val);**

**while(val > balance || (val < 10 && val > 0) || val <=0) //checking if the value bet can be afforded**

**{**

**if(val > balance)**

**printf("You cannot bet more than your balance. Your balance is: %d.\n", balance);**

**else if(val <= 0)**

**printf("You must make a bet. Enter a valid bet amount.\n");**

**else**

**printf("Invalid entry. Try again.\n");**

**scanf("%d", &val);**

**}**

**balance -= val;**

**printf("You've made a bet of %d. Good luck!\n", val);**

**pot = val;**

**}**

**void initdeck()**

**{**

**int i=0;**

**for(i;i<52;i++)**

**{**

**deck[i] = i+1; //assigning values to the cards 1234...52**

**}**

**}**

**void shuffle()**

**{**

**srand(time(NULL));**

**int i=0;**

**for(i;i<52;i++)**

**{**

**sdeck[i] = deck[i]; //taking values for starting deck**

**}**

**i=0;**

**for(i=CARDS-1;i>0;i--)**

**{**

**int j = rand()%(i+1); //randomly selecting card value and putting in starting deck**

**int n = sdeck[i];**

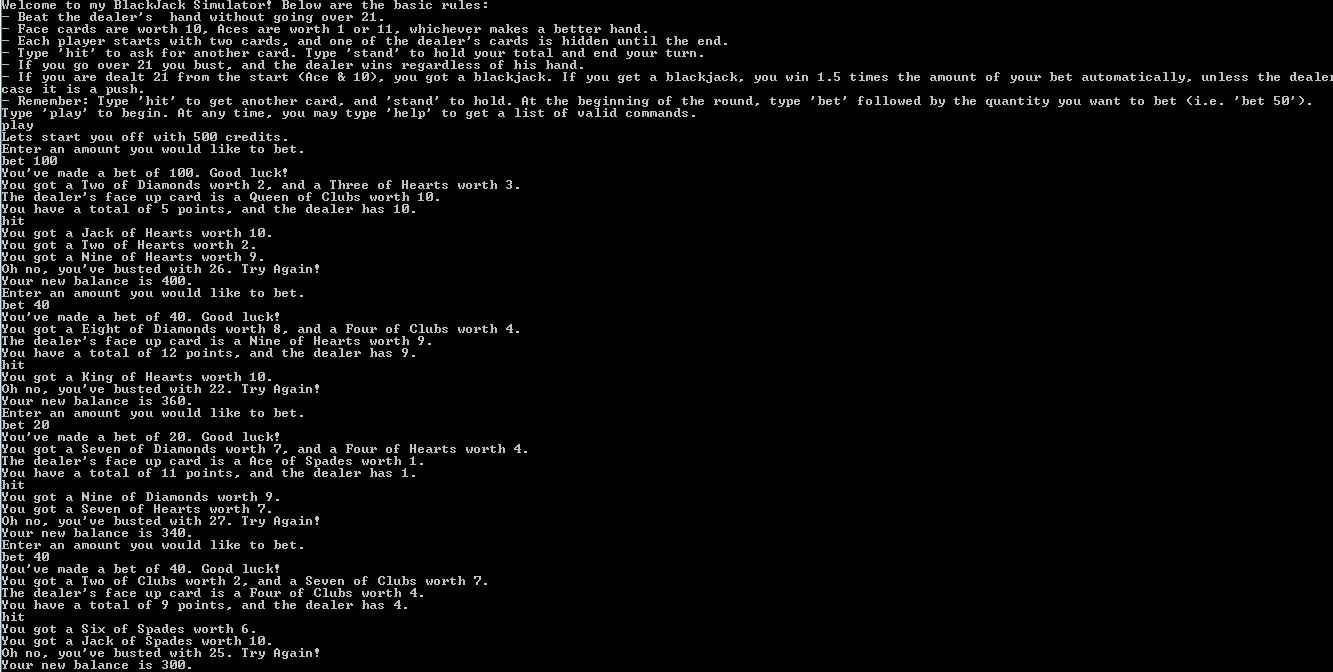
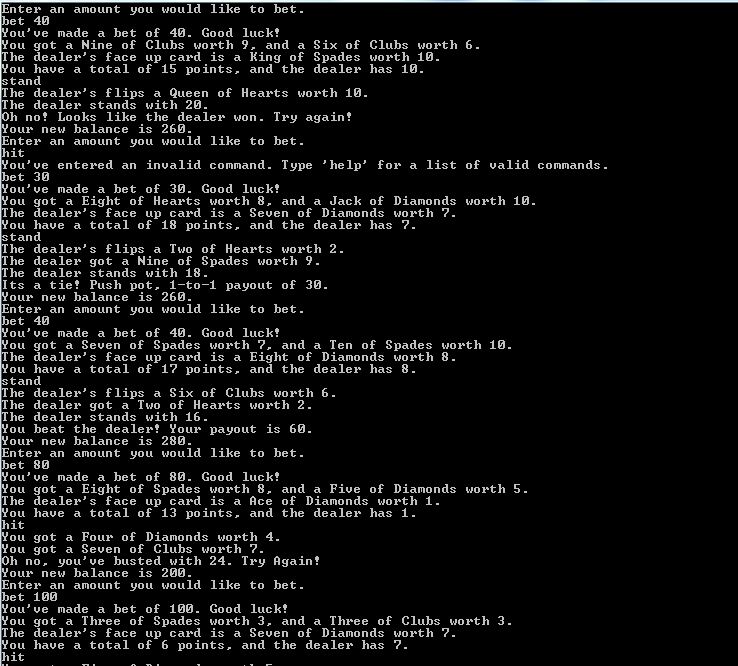
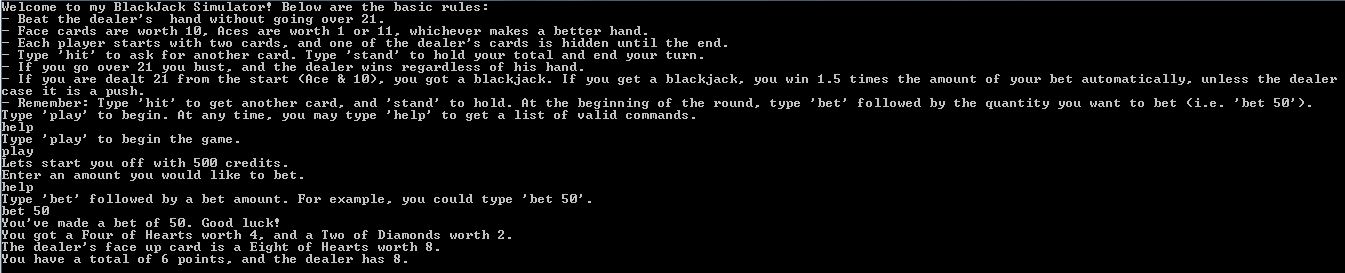
**sdeck[i] = sdeck[j];**

**sdeck[j] = n;**

**}**

**}**

OUTPUT SCREENS

Thank you

Project by:

SAATWIK BISARIA

IT(M) 42711503117