10-2-2022

Practica 1 Black Jack

Materia: Seminario de estructura de datos 1

Sección: D13.

Código: 216584703

Carrera: Ingeniería en computación.

Nombre alumno: Padilla Pérez Jorge Daray

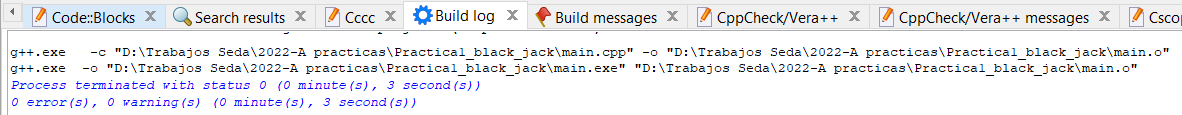
Nombre profesor: Julio Esteban Valdes Lopez



**Introducción**

En esta practica se realizó el juego Black Jack a medias ya que me falto una correcta eliminación de los arreglos del jugador y la máquina, mas que nada por falta de tiempo, obviamente no se pudo completar el juego, pero el sistema para barajear la mano, el repartir las cartas, que la maquina pida o no cartas cuando debe y que el AS sea 11 u 1 como se pidió se logró.

Además falta solucionar algunos bugs con la eliminación que estoy seguro no se hace correctamente y debo de encontrar otra forma de solucionarlo, también falta la implementación del sistema de monedas que no le hayo mayor complejidad más que restar si se pierde o sumar si se gana, y el inicializar ya esta implementado para que no haya problema si se quiere volver a empezar.



En primera no marca errores el programa.

Texto

Descripción generada automáticamente

Una vez ejecutado se aprecia el menú, todo funciona menos el eliminar juego.

Texto

Descripción generada automáticamente

Ya barajeado e impreso el juego se aprecia que esta bien revuelto y se hizo de tal forma que no se repita.

Texto

Descripción generada automáticamente

Se reparte las cartas y se imprime el juego, como todavía sigue en pruebas imprimo las 2 cartas de la compu para ver que funcione bien.

Texto

Descripción generada automáticamente

En este caso como voy ganando le puse que no quiero otra carta entonces la maquina pidió una al ir perdiendo, como se ve en nuestro contador head, utilizado para tener la cabeza de la baraja de 52 para no dar cartas repetidas etc.

Texto

Descripción generada automáticamente

En este caso la compu saco un 5 por lo cual sigue perdiendo y tendrá que pedir otra carta.

Texto

Descripción generada automáticamente

Lastimosamente para la compu le salió un rey y su suma da 28 por ende ganamos y arroja el mensaje.

Texto

Descripción generada automáticamente

Como dije a la hora de borrar tengo problemas y esto pasa, se tiene que solucionar.

Texto

Descripción generada automáticamente

Iniciamos otro juego y aquí la maquina ganara ya que suma 21 su partida inicial, y yo sumo 14.

Texto

Descripción generada automáticamente

Gano la maquina con una suma de 21 a su favor.

Texto

Descripción generada automáticamente

Aquí esta otro caso del juego.

Texto

Descripción generada automáticamente

Voy ganando le pongo que no.

Texto

Descripción generada automáticamente

Sigo ganando le pongo que no.

Texto

Descripción generada automáticamente

Sigo ganando le pongo que no.

Texto

Descripción generada automáticamente

Por fin gane.

**Conclusión**

Me falta todavía solucionar los problemas que se presentan a la hora de querer borrarlo, por ende, no está acabado, siento que le dedique menos tiempo del que tenía que dedicarle, además de que no esta tan sencillo solucionar los pequeños problemas que van surgiendo a la hora de irlo haciendo.

También siento que implemente bien algunas cosas y otras por lo visto pudieron ser mucho más fáciles y entendibles su interpretación, me gustaría poder dedicarle mas tiempo al programa para acabarlo lo antes posible, y no atrasarme con la primer practica que es esta, además siento que con ver o escuchar la lógica que mis compañeros usen para solucionar los problemas me serviría mucho.

**Codigo fuente**

#include <iostream>

#include <stdlib.h>

#include <time.h>

#include <string.h>

#define Maxarreglo 52

#define Maxjugador 11

**using** **namespace** std;

**int** Arreglo\_baraja [Maxarreglo], Arreglo\_barajaj [Maxarreglo], Arreglo\_barajac [Maxarreglo];

**int** valor\_carta, contador = **0**, contadorj = **0**, contadorc = **0**,contador\_cartasjuego = **0**;

**int** head = **0**, headj = **0**, headc = **0**;

**int** dinero = **100**;

**int** totaldj = **0**,totaldc = **0**, totaldj2 = **0**, totaldc2 = **0**;

**struct** Black\_jack{

**char** name[Maxarreglo];

**char** name\_jugador[Maxjugador];

**char** name\_compu[Maxjugador];

}Arreglo\_jugador [Maxjugador], Arreglo\_compu [Maxjugador], Arreglo\_baraja2[Maxarreglo];

**bool** **validar**(**int** num, **int** pos)

{

**for**(**int** i = head; i < pos; i ++)

{

**if** (num == Arreglo\_baraja[i])

{

**return** true;

}

}

**return** false;

}

**void** **random**()

{

**int** num;

srand(time(NULL));

**for**(**int** i = head; i < Maxarreglo; i ++)

{

num = rand()%**52**;

**while** (validar(num, i) == true)

{

num = rand()%**52**;

}

Arreglo\_baraja[i] = num;

}

}

**void** **inicializa**()

{

random();

**for**(**int** i = head; i < Maxarreglo; i ++){

**switch**(Arreglo\_baraja[i]){

**case** **0**: strcpy(Arreglo\_baraja2[i].name, "AS"); valor\_carta = **1**; Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **1**: strcpy(Arreglo\_baraja2[i].name, "AS"); valor\_carta = **1**; Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **2**: strcpy(Arreglo\_baraja2[i].name, "AS"); valor\_carta = **1**; Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **3**: strcpy(Arreglo\_baraja2[i].name, "AS"); valor\_carta = **1**; Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **4**: strcpy(Arreglo\_baraja2[i].name, "2"); valor\_carta = **2**; Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **5**: strcpy(Arreglo\_baraja2[i].name, "2"); valor\_carta = **2**; Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **6**: strcpy(Arreglo\_baraja2[i].name, "2"); valor\_carta = **2**; Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **7**: strcpy(Arreglo\_baraja2[i].name, "2"); valor\_carta = **2**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **8**: strcpy(Arreglo\_baraja2[i].name, "3"); valor\_carta = **3**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **9**: strcpy(Arreglo\_baraja2[i].name, "3"); valor\_carta = **3**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **10**: strcpy(Arreglo\_baraja2[i].name, "3"); valor\_carta = **3**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **11**: strcpy(Arreglo\_baraja2[i].name, "3"); valor\_carta = **3**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **12**: strcpy(Arreglo\_baraja2[i].name, "4"); valor\_carta = **4**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **13**: strcpy(Arreglo\_baraja2[i].name, "4"); valor\_carta = **4**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **14**: strcpy(Arreglo\_baraja2[i].name, "4"); valor\_carta = **4**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **15**: strcpy(Arreglo\_baraja2[i].name, "4"); valor\_carta = **4**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **16**: strcpy(Arreglo\_baraja2[i].name, "5"); valor\_carta = **5**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **17**: strcpy(Arreglo\_baraja2[i].name, "5"); valor\_carta = **5**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **18**: strcpy(Arreglo\_baraja2[i].name, "5"); valor\_carta = **5**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **19**: strcpy(Arreglo\_baraja2[i].name, "5"); valor\_carta = **5**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **20**: strcpy(Arreglo\_baraja2[i].name, "6"); valor\_carta = **6**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **21**: strcpy(Arreglo\_baraja2[i].name, "6"); valor\_carta = **6**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **22**: strcpy(Arreglo\_baraja2[i].name, "6"); valor\_carta = **6**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **23**: strcpy(Arreglo\_baraja2[i].name, "6"); valor\_carta = **6**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **24**: strcpy(Arreglo\_baraja2[i].name, "7"); valor\_carta = **7**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **25**: strcpy(Arreglo\_baraja2[i].name, "7"); valor\_carta = **7**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **26**: strcpy(Arreglo\_baraja2[i].name, "7"); valor\_carta = **7**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **27**: strcpy(Arreglo\_baraja2[i].name, "7"); valor\_carta = **7**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **28**: strcpy(Arreglo\_baraja2[i].name, "8"); valor\_carta = **8**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **29**: strcpy(Arreglo\_baraja2[i].name, "8"); valor\_carta = **8**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **30**: strcpy(Arreglo\_baraja2[i].name, "8"); valor\_carta = **8**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **31**: strcpy(Arreglo\_baraja2[i].name, "8"); valor\_carta = **8**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **32**: strcpy(Arreglo\_baraja2[i].name, "9"); valor\_carta = **9**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **33**: strcpy(Arreglo\_baraja2[i].name, "9"); valor\_carta = **9**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **34**: strcpy(Arreglo\_baraja2[i].name, "9"); valor\_carta = **9**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **35**: strcpy(Arreglo\_baraja2[i].name, "9"); valor\_carta = **9**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **36**: strcpy(Arreglo\_baraja2[i].name, "10"); valor\_carta = **10**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **37**: strcpy(Arreglo\_baraja2[i].name, "10"); valor\_carta = **10**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **38**: strcpy(Arreglo\_baraja2[i].name, "10"); valor\_carta = **10**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **39**: strcpy(Arreglo\_baraja2[i].name, "10"); valor\_carta = **10**;Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **40**: strcpy(Arreglo\_baraja2[i].name, "Joto"); valor\_carta = **10**; Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **41**: strcpy(Arreglo\_baraja2[i].name, "Joto"); valor\_carta = **10**; Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **42**: strcpy(Arreglo\_baraja2[i].name, "Joto"); valor\_carta = **10**; Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **43**: strcpy(Arreglo\_baraja2[i].name, "Joto"); valor\_carta = **10**; Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **44**: strcpy(Arreglo\_baraja2[i].name, "Quenn"); valor\_carta = **10**; Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **45**: strcpy(Arreglo\_baraja2[i].name, "Quenn"); valor\_carta = **10**; Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **46**: strcpy(Arreglo\_baraja2[i].name, "Quenn"); valor\_carta = **10**; Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **47**: strcpy(Arreglo\_baraja2[i].name, "Quenn"); valor\_carta = **10**; Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **48**: strcpy(Arreglo\_baraja2[i].name, "K"); valor\_carta = **10**; Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **49**: strcpy(Arreglo\_baraja2[i].name, "K"); valor\_carta = **10**; Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **50**: strcpy(Arreglo\_baraja2[i].name, "K"); valor\_carta = **10**; Arreglo\_baraja[i] = valor\_carta; **break**;

**case** **51**: strcpy(Arreglo\_baraja2[i].name, "K"); valor\_carta = **10**; Arreglo\_baraja[i] = valor\_carta; **break**;

}

cout <<"valor de carta:"<<Arreglo\_baraja[i]<<endl;

contador ++;

}

cout <<"contador: "<<contador<<endl;

}

**void** **prueba**()

{

**for**(**int** i = head; i < Maxarreglo; i ++){

cout<<" posicion :"<<i<<endl;

cout <<"Cartas :"<<Arreglo\_baraja2[i].name<<endl;

}

cout <<"contador: "<<contador<<endl;

}

**void** **repartir\_cartasi**()

{

**if** (contador < **4**)

{

cout <<"Ya se acabaron las cartas"<<endl;

**return**;

}

**for** (**int** i = head; i < **4**; i++)

{

**if** (i < **2**)

{

strcpy(Arreglo\_jugador[i].name\_jugador, Arreglo\_baraja2[i].name);

Arreglo\_barajaj[i] = Arreglo\_baraja[i];

contador\_cartasjuego++;

contadorj++;

}

**if** (i >= **2**)

{

strcpy(Arreglo\_compu[i-**2**].name\_compu, Arreglo\_baraja2[i].name);

Arreglo\_barajac[i-**2**] = Arreglo\_baraja[i];

contador\_cartasjuego++;

contadorc++;

}

contador--;

head++;

}

}

**void** **pedir\_cartasc**()

{

**if**(totaldj > totaldc)

{

**for** (**int** i = head; i < head+**1**; i++)

{

strcpy(Arreglo\_compu[i-**2**].name\_compu, Arreglo\_baraja2[i].name);

Arreglo\_barajac[i-**2**] = Arreglo\_baraja[i];

contador\_cartasjuego++;

contadorc++;

contador--;

}

head++;

}

**else**{

**if**(totaldj2 > totaldc2)

{

**for** (**int** i = head; i < head+**1**; i++)

{

strcpy(Arreglo\_compu[i-**2**].name\_compu, Arreglo\_baraja2[i].name);

Arreglo\_barajac[i-**2**] = Arreglo\_baraja[i];

contador\_cartasjuego++;

contadorc++;

contador--;

}

head++;

}

}

}

**void** **eliminar\_juegos**()

{

**for** (**int** i = **0**; i < contadorj ; i++){

Arreglo\_barajaj[i] = Arreglo\_barajaj[i + **1**];

headj --;

}

**for** (**int** i = **0**; i < contadorc ; i++){

Arreglo\_barajac[i] = Arreglo\_barajac[i + **1**];

headc--;

}

}

**void** **comprobar\_win**()

{

**if** (totaldj == **21**)

{

cout << "ganaste"<<endl;

system("pause");

eliminar\_juegos();

**return**;

}

**if** (totaldj > **21**)

{

cout << "gano la maquina"<<endl;

system("pause");

eliminar\_juegos();

**return**;

}

**if** (totaldc == **21**)

{

cout << "gano la maquina"<<endl;

system("pause");

eliminar\_juegos();

**return**;

}

**if** (totaldc > **21**)

{

cout << "ganaste"<<endl;

system("pause");

eliminar\_juegos();

**return**;

}

**if** (totaldj2 == **21**)

{

cout << "ganaste"<<endl;

system("pause");

eliminar\_juegos();

**return**;

}

**if** (totaldj2 > **21**)

{

cout << "gano la maquina"<<endl;

system("pause");

eliminar\_juegos();

**return**;

}

**if** (totaldc2 == **21**)

{

cout << "gano la maquina"<<endl;

system("pause");

eliminar\_juegos();

**return**;

}

**if** (totaldc2 > **21**)

{

cout << "ganaste"<<endl;

system("pause");

eliminar\_juegos();

**return**;

}

}

**void** **pedir\_cartas**()

{

**char** letra;

**int** c = **2**;

cout <<"Quieres otra carta ? pon 's' para si o 'n' para no: "<<endl;cin >> letra;

**if** (letra == 's')

{

**for** (**int** i = head; i < head+**1**; i++)

{

strcpy(Arreglo\_jugador[i-c].name\_jugador, Arreglo\_baraja2[i].name);

Arreglo\_barajaj[i-c] = Arreglo\_baraja[i];

contador\_cartasjuego++;

contadorj++;

contador--;

}

head++;

}

**else**{

**return**;

}

}

**void** **imprimir\_juego**()

{

**int** nuevo\_valor = **11**;

**if** (contador\_cartasjuego < **5**)

{

**int** total = **0**, totalc = **0**;

**for**(**int** i = headj; i < contadorj; i ++){

cout<<"Cartas jugador: "<<Arreglo\_jugador[i].name\_jugador<<endl;

**if** (Arreglo\_barajaj[i] == **1** && total < **21**)

{

**if** (Arreglo\_barajaj[i] == **1** && total == **11**)

{

comprobar\_win();

**return**;

}

**else**{

**if**(Arreglo\_barajaj[i] == **1** && total < **21**)

{

Arreglo\_barajaj[i] = nuevo\_valor;

}

}

}

total = total + Arreglo\_barajaj[i];

}

cout << "Total suma jugador: "<<total<<endl;

totaldj = total;

comprobar\_win();

**for**(**int** i = headc; i < contadorc; i ++){

cout<<"Cartas compu: "<<Arreglo\_compu[i].name\_compu<<endl;

**if** (Arreglo\_barajac[i] == **1** && totalc < **21**)

{

**if** (Arreglo\_barajac[i] == **1** && totalc == **11**)

{

comprobar\_win();

**return**;

}

**else**{

**if**(Arreglo\_barajac[i] == **1** && totalc < **21**)

{

Arreglo\_barajac[i] = nuevo\_valor;

}

}

}

totalc = totalc + Arreglo\_barajac[i];

}

totaldc = totalc;

cout << "Total suma compu: "<<totalc<<endl;

pedir\_cartasc();

comprobar\_win();

}

**else**{

**int** total = **0**, totalc = **0**, totaldj = **0**,totaldc = **0**;

**for**(**int** i = headj; i < contadorj; i ++){

cout<<"Cartas jugador: "<<Arreglo\_jugador[i].name\_jugador<<endl;

**if** (Arreglo\_barajaj[i] == **1** && total < **21**)

{

**if** (Arreglo\_barajaj[i] == **1** && total == **11**)

{

comprobar\_win();

**return**;

}

**else**{

**if**(Arreglo\_barajaj[i] == **1** && total < **21**)

{

Arreglo\_barajaj[i] = nuevo\_valor;

}

}

}

total = total + Arreglo\_barajaj[i];

}

cout << "Total suma jugador: "<<total<<endl;

totaldj2 = total;

comprobar\_win();

**for**(**int** i = headc; i < contadorc; i ++){

cout<<"Cartas compu: "<<Arreglo\_compu[i].name\_compu<<endl;

**if** (Arreglo\_barajac[i] == **1** && totalc < **21**)

{

**if** (Arreglo\_barajac[i] == **1** && totalc == **11**)

{

comprobar\_win();

**return**;

}

**else**{

**if**(Arreglo\_barajac[i] == **1** && totalc < **21**)

{

Arreglo\_barajac[i] = nuevo\_valor;

}

}

}

totalc = totalc + Arreglo\_barajac[i];

}

totaldc2 = totalc;

cout << "Total suma compu: "<<totalc<<endl;

pedir\_cartasc();

comprobar\_win();

}

pedir\_cartas();

}

**int** **main**(){

**int** opcion;

**while**(opcion!=**6**){

cout<<"**\n** Que opcion deseas, **\n**6-Salir **\n**1-barajear cartas **\n**2-imprime cartas barajeadas **\n**3-repartir cartas **\n**4-imprimir juego ";

cout<<"**\n**5-eliminar juego **\n**Opcion: "<<endl;

cin>>opcion;

**switch**(opcion){

**case** **1**: system("cls");

inicializa();

cout<<" head :"<<head<<endl;

**break**;

**case** **2**: system("cls");

prueba();

cout<<" head :"<<head<<endl;

**break**;

**case** **3**: system("cls");

repartir\_cartasi();

cout<<" head :"<<head<<endl;

**break**;

**case** **4**: system("cls");

imprimir\_juego();

cout<<" head :"<<head<<endl;

**break**;

**case** **5**: system("cls");

eliminar\_juegos();

cout<<" head :"<<head<<endl;

**break**;

**case** **6**: **break**;

**default:** printf("Esa opcion no existe...**\n**");

}

}

**return** **0**;

}