Análisis de Sistemas 2020

 K1004

MIERCOLES-MAÑANA

ALGORITMOS Y ESTRUCTURA DE DATOS

ING. PABLO D. MENDEZ

|  |
| --- |
| Grupo |

|  |  |
| --- | --- |
| Nombre y Apellido | Legajo |
| ALEJANDRA CHOQUE QUISPE | 1730988 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

|  |  |
| --- | --- |
| Fecha de Presentación |  |
| Fecha de Devolución |  |
| Calificación |  |
| Firma Profesor |  |

Universidad Tecnológica Nacional

Facultad Regional Buenos Aires

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <iostream>

#include <conio.h>

#include <direct.h>

#include <stdio.h>

#define ARCHIVOCUENTAS "abc/Cuentas.Bic"

#define ARCHIVOLOTESPROCESADOS "abc/Procesados.Bic"

#define ARCHIVOAUX "abc/auxiliar"

using namespace std;

bool cargar\_cuenta\_bancaria();

char menucuentas();

void cargarMovimientos();

int desactivarCuenta();

void procesarLotes(int contar);

void mostrarlote();

int buscarCuentaPorNumeroDeCliente(int NroDeCliente);

int buscarTarjeta(int buscar,int opcion);

void borrarCuentasInactivas();

void ordenarCuentasPorSaldoDescendiente();

int numeroDeMovimientos(int aux);

bool verificarSaldo( float SaldoAExtraer,int IDtarjeta);

struct Tarjeta

{

int TarjetaID;

int fechaCreacion;

int Activa=0;

float Saldo;

int NroCliente;

};

struct Moviento

{

int movientoID;

int fecha;

int Hora;

float monto;

int tarjetaID;

};

char menucuentas()// MENU

{

char opcion;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* MENU \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\* DIGITE UNA OPCION \*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<endl;

cout<<" 1- CARGAR UNA NUEVA CUENTA."<<endl;

cout<<" 2- DESACTIVAR CUENTA." <<endl;

cout<<" 3- BUSCAR CLIENTE O TARJETA. "<<endl;

cout<<" 4- LISTAR CUENTA POR SALDO DESCENDIENTE. "<<endl;

cout<<" 5- CREAR UN LOTE DE MOVIMIENTO. "<<endl;

cout<<" 6- FINALIZAR JORNADA. "<<endl;

cout<<endl<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

do

{

opcion=getch();

}while(!(opcion>='1' && opcion<='7') && (opcion!='27'));

switch(opcion)

{

case'1':

//CREAR TARJETA

system("cls");

if(cargar\_cuenta\_bancaria())

{

cout<<endl;

cout<<" CUENTA ACTIVADA EXITOSAMENTE! "<<endl;

cout<<endl;

cout<<" Presione cualquier tecla para continuar... "<<endl;

getch();

system("cls");

}

else

{

cout<<endl;

printf(" NUMERO DE TARJETA EXISTENTE EN EL SISTEMA!\n ");

cout<<endl;

system("pause");

system("cls");

}

break;

case '2':

// DESACTIVAR TARJETA

system("cls");

desactivarCuenta();

system("cls");

break;

case '3':

{

//BUSCO TARJETA O NUMERO DE CLIENTE

system("cls");

char buscar=0;

cout<<"\*\*\*\*\*\*\*\*\* DIGITE UNA OPCION \*\*\*\*\*\*\*\*"<<endl;

cout<<endl;

cout<<" 1 - TARJETAID. "<<endl;

cout<<" 2 - NUMERO DEL CLIENTE . "<<endl;

do

{

cin.clear();

buscar=getch();

}while(buscar!='1' && buscar !='2');

system("cls");

if( buscar=='1')

{

buscar=0;

int tarjetaid=0;

int cuentaNro=0;

cout<<" \*\*\*\*\*\*\*\*\*\*\* DIGITE EL NUMERO DE LA TARJETA \*\*\*\*\*\*\*\*\*\* "<<endl;

cout<<endl;

cout<<" TARJETA ID:"<<endl;

cin>>tarjetaid;

//BUSCO TARJETA

buscar=buscarTarjeta(tarjetaid,1);

if(buscar==0)

{

cout<<endl;

cout<<" TARJETA INEXISTENTE EN EL SISTEMA! "<<endl;

cout<<endl;

system("pause");

}

else

{

cout<<endl;

system("pause");

}

}

else

{

//BUSCO NUMERO DE CLIENTE

int NroDeCliente;

cout<<"\*\*\*\*\*\*\* DIGITE EL NUMERO DEL CLIENTE \*\*\*\*\*\*\*\* "<<endl;

cout<<endl;

cout<<" NUMERO DEL CLIENTE:"<<endl;

cin>>NroDeCliente;

if(!(buscar=buscarCuentaPorNumeroDeCliente(NroDeCliente)))

{

cout<<" CLIENTE INEXISTENTE EN EL SISTEMA! "<<endl;

cout<<endl;

system("pause");

}

else

{

cout<<endl;

system("pause");

}

}

}

system("cls");

break;

case '4' :

// ORDENO CUENTAS POR SALDO DESCENDIENTE

system("cls");

ordenarCuentasPorSaldoDescendiente();

break;

case'5':

{

//MOVIMENTAR CUENTAS

system("cls");

char opcionLote=0;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* DIGITE UNA OPCION \*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<endl;

cout<<" 1- MOVIMIENTAR CUENTAS."<<endl;

cout<<" 2- VISUALIZAR MOVIMIENTOS . "<<endl;

do

{

cin.clear();

opcionLote=getch();

}while(opcionLote!='1' && opcionLote !='2');

system("cls");

if( opcionLote=='1')

{

cargarMovimientos();

}

else

{

mostrarlote();

}

}

system("cls");

break;

case '6':

system("cls");

{

char borrador=0;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\* CONFIRMAR OPCION \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<endl;

cout<<" 1 - BORRAR CUENTAS INACTIVAS. "<<endl;

cout<<" 0 - VOLVER AL MENU PRINCIPAL. "<<endl;

do{

cin.clear();

borrador=getch();

}while(borrador!='1' && borrador!='0');

system("cls");

if(borrador=='1')

{

borrarCuentasInactivas();

remove(ARCHIVOLOTESPROCESADOS);

break;

}

else

{

opcion=1;

system("cls");

break;

}

}

}

return opcion;

}

int numeroDeMovimientos() //NUMERO DEL MOVIMIENTO ACTUAL

{

FILE\*movimientosAux;

Moviento aux1;

int tam;

//CALCULO EL NUMERO DE LOTES

if(movimientosAux=fopen(ARCHIVOLOTESPROCESADOS,"rb"))

{

fseek(movimientosAux,0,SEEK\_END);

tam=(ftell(movimientosAux))/sizeof(aux1);

fclose(movimientosAux);

}

return tam;

}

void borrarCuentasInactivas() //BORRADO DE LAS CUENTAS INACTIVAS

{

FILE\*cuentasBancariasCarga;

int cont=0;

FILE\*archivoAux;

Tarjeta aux1;

archivoAux=fopen(ARCHIVOAUX,"wb");

cuentasBancariasCarga=fopen(ARCHIVOCUENTAS,"rb+");

if(cuentasBancariasCarga==NULL)

{

cout<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\* JORNADA FINALIZADA \*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<endl;

cout<<" NO SE REGISTRA NINGUNA CUENTA EN EL SISTEMA! "<<endl;

cout<<endl;

cout<<" Presione cualquier tecla para continuar..."<<endl;

getch();

system("cls");

}

else

{

fread(&aux1,sizeof(Tarjeta),1,cuentasBancariasCarga);

while(!feof(cuentasBancariasCarga))

{

cont=1;

if(aux1.Activa==1)

{

fwrite(&aux1,sizeof(Tarjeta),1,archivoAux);

}

fread(&aux1,sizeof(Tarjeta),1,cuentasBancariasCarga);

}

fclose(cuentasBancariasCarga);

fclose(archivoAux);

archivoAux=fopen(ARCHIVOAUX,"rb");

cuentasBancariasCarga=fopen(ARCHIVOCUENTAS,"wb");

if(cuentasBancariasCarga==NULL)

{

cout<<" NO SE PUEDE ABRIR ARCHIVO! "<<endl;

}

fread(&aux1,sizeof(Tarjeta),1,archivoAux);

while(!feof(archivoAux))

{

if(aux1.Activa==1)

{

fwrite(&aux1,sizeof(Tarjeta),1,cuentasBancariasCarga);

}

fread(&aux1,sizeof(Tarjeta),1,archivoAux);

}

fclose(cuentasBancariasCarga);

fclose(archivoAux);

if(cont)

{

cout<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\* JORNADA FINALIZADA \*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<endl;

cout<<" ACTUALIZACION EXITOSA! "<<endl;

cout<<endl;

cout<<endl;

cout<<" Presione cualquier tecla para continuar..."<<endl;

}

else

{

cout<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\* JORNADA FINALIZADA \*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<endl;

cout<<" ERROR AL INTENTAR A ACTUALIZAR! "<<endl;

cout<<endl;

cout<<" Presione cualquier tecla para continuar..."<<endl;

}

remove(ARCHIVOLOTESPROCESADOS);

getch();

system("cls");

}

}

void actualizarSaldo(Moviento &aux) //ACTUALIZO SALDO DE LA CUENTA CUANDO SE INGRESA EN EL MOVIMIENTO

{

FILE\*cuentasBancariasCarga;

Tarjeta auxiliar;

cuentasBancariasCarga=fopen(ARCHIVOCUENTAS,"rb+");

if(cuentasBancariasCarga==NULL)

{

cout<<" ARCHIVO DE CUENTAS INEXISTENTE EN EL SISTEMA! "<<endl;

}

fread(&auxiliar,sizeof(Tarjeta),1,cuentasBancariasCarga);

while(!feof(cuentasBancariasCarga))

{

if(aux.tarjetaID==auxiliar.TarjetaID)

{

auxiliar.Saldo=auxiliar.Saldo+aux.monto;

fseek(cuentasBancariasCarga,(-1)\*sizeof(Tarjeta),SEEK\_CUR);

fwrite(&auxiliar,sizeof(Tarjeta),1,cuentasBancariasCarga);

break;

}

fread(&auxiliar,sizeof(Tarjeta),1,cuentasBancariasCarga);

}

fclose(cuentasBancariasCarga);

}

int buscarCuentaPorNumeroDeCliente(int NroDeCliente)//BUSCO TARJETA POR NUEMERO DE CLIENTE

{

FILE\*cuentasBancariasCarga;

Tarjeta auxiliar;

int cont=0;

int retornar=0;

system("cls");

cout<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* TARJETA(S) DEL CLIENTE \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* "<<endl;

cout<<endl;

cuentasBancariasCarga=fopen(ARCHIVOCUENTAS,"rb");

if(cuentasBancariasCarga==NULL)

{

return retornar;

}

else

{

fread(&auxiliar,sizeof(Tarjeta),1,cuentasBancariasCarga);

while(!feof(cuentasBancariasCarga))

{

int anio=double(auxiliar.fechaCreacion/10000);

int dia=auxiliar.fechaCreacion%100;

int mes=auxiliar.fechaCreacion%10000;

int mes1=mes/100;

if(NroDeCliente==auxiliar.NroCliente)

{

if(auxiliar.Activa==1)

{

cont++;

cout <<" CUENTA NUMERO "<<cont<<endl;

cout<<endl;

cout<<" TARJETA ID FECHA DE CREACION SALDO NRODECLIENTE ESTADO "<<endl;

cout<<" "<<auxiliar.TarjetaID<<" "<<anio<<"/"<<mes1<<"/"<<dia<<" $ "<<auxiliar.Saldo<<" "<<auxiliar.NroCliente<<" ACTIVA."<<endl;

cout<<endl;

retornar=1;

}

else

{

cont++;

cout <<" CUENTA NUMERO "<<cont<<endl;

cout<<" TARJETA ID FECHA DE CREACION SALDO NRODECLIENTE ESTADO "<<endl;

cout<<" "<<auxiliar.TarjetaID<<" "<<anio<<"/"<<mes1<<"/"<<dia<<" $ "<<auxiliar.Saldo<<" "<<auxiliar.NroCliente<<" INACTIVA."<<endl;

cout<<endl;

retornar=1;

}

}

fread(&auxiliar,sizeof(Tarjeta),1,cuentasBancariasCarga);

}

fclose(cuentasBancariasCarga);

}

return retornar;

}

int buscarTarjeta( int buscar, int opcion) //BUSCO TARJETA POR NUMERO DE CLIENTE

{

FILE\*cuentasBancariasCarga;

Tarjeta auxiliar;

int retornar=0;

cuentasBancariasCarga=fopen(ARCHIVOCUENTAS,"rb");

if(cuentasBancariasCarga==NULL)

{

if(opcion==0)

{

cout<<endl;

cout<<" TARJETA INEXISTENTE EN EL SISTEMA! "<<endl;

cout<<endl;

system("pause");

system("cls");

return retornar;

}

return retornar;

}

fread(&auxiliar,sizeof(Tarjeta),1,cuentasBancariasCarga);

int contador;

while(!feof(cuentasBancariasCarga))

{

if(buscar==auxiliar.TarjetaID )

{

if(opcion==4)

{

return retornar=1;

break;

}

if(opcion==0 && auxiliar.Activa!=0)

{

return retornar=1;

break;

}

if(opcion==2 && auxiliar.Activa!=0)

{

return retornar=1;

break;

}

else if (opcion==1 && auxiliar.Activa==0)

{

system("cls");

cout<<endl;

cout<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* DATOS TARJETA \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* "<<endl;

cout<<endl;

cout<<" TARJETA ID FECHA DE CREACION SALDO NRODECLIENTE ESTADO. "<<endl;

cout<<" "<<auxiliar.TarjetaID<<" "<<auxiliar.fechaCreacion<<" $ "<<auxiliar.Saldo<<" "<<auxiliar.NroCliente<<" INACTIVA"<<endl;

return retornar=1;

break;

}

else if (opcion==1 && auxiliar.Activa==1)

{

system("cls");

cout<<endl;

cout<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* DATOS TARJETA \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* "<<endl;

cout<<endl;

cout<<" TARJETA ID FECHA DE CREACION SALDO NRODECLIENTE ESTADO. "<<endl;

cout<<" "<<auxiliar.TarjetaID<<" "<<auxiliar.fechaCreacion<<" $ "<<auxiliar.Saldo<<" "<<auxiliar.NroCliente<<" ACTIVA"<<endl;

return retornar=1;

break;

}

}

fread(&auxiliar,sizeof(Tarjeta),1,cuentasBancariasCarga);

}

if(retornar==0 && opcion==0)

{

cout<<endl;

cout<<" TARJETA INEXISTENTE O DESACTIVADA! "<<endl;

cout<<endl;

cout<<"Presione cualquier tecla para continuar..."<<endl;

getch();

return retornar;

}

else if(retornar==0 && opcion==2)

{

return retornar;

}

fclose(cuentasBancariasCarga);

return retornar;

}

bool verificarSaldo(float SaldoAExtraer,int IDtarjeta) //VERIFICO SI LA CUENTA TIENE SALDO PARA SER EXTRAIDO

{

FILE\*cuentasBancariasCarga;

Tarjeta aux;

if(cuentasBancariasCarga=fopen(ARCHIVOCUENTAS,"rb"))

{

while(fread(&aux,sizeof(Tarjeta),1,cuentasBancariasCarga))

{

if(IDtarjeta==aux.TarjetaID )

{

float Saldo=SaldoAExtraer\*(-1);

if(Saldo<aux.Saldo)

{

return true;

}

else

{

return false;

}

}

}

}

}

void cargarMovimientos() // MOVIMIENTOS

{

//MOVIMENTAR CUENTAS

int contar=0;

FILE\*movimientosAux;

Moviento aux;

movimientosAux=fopen(ARCHIVOLOTESPROCESADOS,"ab");

if( movimientosAux==NULL)

{

cout<<" ARCHIVO DE MOVIMIENTOS INEXISTENTE EN EL SISTEMA! "<<endl;

}

char opcion1=1;

while((opcion1==1) || (opcion1=='1'))

{

opcion1=0;

char opcion2=0;

cout <<"\*\*\*\*\*\*\*\*\*\*\* DIGITE UNA OPCION \*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<endl;

cout <<" 0 - RETIRAR DINERO. "<<endl;

cout <<" 1 - INGRESAR DINERO. "<<endl;

do

{

opcion2=getch();

system("cls");

}while( opcion2!='0' && opcion2!='1');

if(movimientosAux=fopen(ARCHIVOLOTESPROCESADOS,"ab"))

{

cout<<" \*\*\*\*\*\*\*\*\*\*\*\* CARGA DE MOVIMIENTOS \*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<endl;

cout <<" NRO DEL MOVIMIENTO "<<endl;

aux.movientoID=numeroDeMovimientos();

cout<<" "<<aux.movientoID<<endl;

cout<<" INGRESE EL NUMERO DE LA TARJETA AAAAA :"<<endl;

cin>>aux.tarjetaID;

int cuentabuscada=aux.tarjetaID;

//BUSCO SABER SI LA CUENTA ESTA CREADA Y ACTIVA EN EL SISTEMA

int poscuenta = buscarTarjeta(cuentabuscada,0);

if(poscuenta)

{

if ( opcion2=='1')

{

//INGRESAR DINERO A LA CUENTA

cout<<" INGRESE EL MONTO $ A CARGAR EN LA CUENTA :"<<aux.tarjetaID<<endl;

cin>>aux.monto;

cout<<" FECHA AAAAMMDD : "<<endl;

cin>>aux.fecha;

cout<<" HORA HHMM :"<<endl;

cin>>aux.Hora;

aux.movientoID++;

//ACTUALIZA SALDO DE LA TARJETA

actualizarSaldo(aux);

fwrite(&aux,sizeof(Moviento),1,movimientosAux);

cout<<endl;

cout<<" MOVIMIENTO REALIZADO CON EXITO!"<<endl;

cout<<endl;

system("pause");

system("cls");

}

else

{

// EXTRAER DINERO DE LA CUENTA

cout<<" INGRESE EL MONTO $ A RETIRAR DE LA CUENTA :"<<aux.tarjetaID<<endl;

cin>>aux.monto;

aux.monto=aux.monto\*(-1);

if(verificarSaldo( aux.monto,aux.tarjetaID))

{

cout<<" FECHA AAAAMMDD : "<<endl;

cin>>aux.fecha;

cout<<" HORA HHMM :"<<endl;

cin>>aux.Hora;

aux.movientoID++;

//ACTUALIZO SALDO EN LA CUENTA

actualizarSaldo(aux);

fwrite(&aux,sizeof(Moviento),1,movimientosAux);

cout<<endl;

cout<<" MOVIMIENTO REALIZADO CON EXITO!"<<endl;

cout<<endl;

system("pause");

system("cls");

}

else

{

cout<<endl;

cout<<" MONTO INDISPONIBLE EN LA CUENTA. "<<endl;

cout<<endl;

system("pause");

system("cls");

}

}

}

fclose(movimientosAux); //1 LOTE GRABADO

}

system("cls");

//CIERO ARCHIVO PROCESADOS

do

{

cout<<endl;

cout<<" DESEAS SEGUIR MOVIMENTANDO CUENTAS? "<<endl;

cout<<endl;

cout<<" 0 - NO "<<endl;

cout<<" 1 - SI "<<endl;

cin.clear();

opcion1=getch();

system("cls");

}

while(opcion1!='1' && opcion1!='0');

}

fclose(movimientosAux); //1 LOTE GRABADO

}

void ordenarCuentasPorSaldoDescendiente() // ORDENAMIENTO

{

FILE\*cuentasBancariasCarga;

int tam=0,i=0;

Tarjeta auxiliar;

if(cuentasBancariasCarga=fopen(ARCHIVOCUENTAS,"rb"))

{

fseek(cuentasBancariasCarga,0,SEEK\_END);

tam=(ftell(cuentasBancariasCarga))/sizeof(auxiliar);

fclose(cuentasBancariasCarga);

}

Tarjeta vc[tam];

cout<<"\*\*\*\*\*\*\*\* CUENTAS ORDENADAS EN SALDO DESCENDIENTE \*\*\*\*\*\*\*\*"<<endl;

cout<<endl;

if(cuentasBancariasCarga=fopen(ARCHIVOCUENTAS,"rb"))

{

fread(&auxiliar,sizeof(Tarjeta),1,cuentasBancariasCarga);

while(!feof(cuentasBancariasCarga))

{

// voy copiando a las posiciones del vector estructura

//cada elemento del archivo.

vc[i].NroCliente=auxiliar.NroCliente;

vc[i].TarjetaID=auxiliar.TarjetaID;

vc[i].Activa=auxiliar.Activa;

vc[i].fechaCreacion=auxiliar.fechaCreacion;

vc[i].Saldo=auxiliar.Saldo;

i++;

fread(&auxiliar,sizeof(Tarjeta),1,cuentasBancariasCarga);

}

fclose(cuentasBancariasCarga);

}

else

{

cout<<endl;

cout<<" ARCHIVO DE CUENTAS INEXISTENTE!"<<endl;

cout<<endl;

}

//TODOS LOS ELEMENTOS ESTAN COPIADOS AL VECTOR.

for(int j=0;j<tam-1;j++)

{

for(int k=0;k<tam-1-j;k++)

{

if(vc[k+1].Saldo > vc[k].Saldo)

{

// ORDENAMIENTO DE LAS CUENTAS EN SALDOS DESCENDIENTES

auxiliar.NroCliente=vc[k].NroCliente;

auxiliar.TarjetaID=vc[k].TarjetaID;

auxiliar.fechaCreacion=vc[k].fechaCreacion;

auxiliar.Saldo=vc[k].Saldo;

auxiliar.Activa=vc[k].Activa;

vc[k].NroCliente=vc[k+1].NroCliente;

vc[k].TarjetaID=vc[k+1].TarjetaID;

vc[k].fechaCreacion=vc[k+1].fechaCreacion;

vc[k].Saldo=vc[k+1].Saldo;

vc[k].Activa=vc[k+1].Activa;

vc[k+1].NroCliente=auxiliar.NroCliente;

vc[k+1].TarjetaID=auxiliar.TarjetaID;

vc[k+1].fechaCreacion=auxiliar.fechaCreacion;

vc[k+1].Saldo=auxiliar.Saldo;

vc[k+1].Activa=auxiliar.Activa;

}

}

}

for( int m=0;m<tam;m++)

{

cout<<"-------------------------------"<<endl;

cout<<"Tarjeta ID:"<<vc[m].TarjetaID<<endl;

cout<<"NroCliente :"<<vc[m].NroCliente<<endl;

cout<<"Fecha de Creacion :"<<vc[m].fechaCreacion<<endl;

cout<<"Saldo $:"<<vc[m].Saldo<<endl;

if(vc[m].Activa==1)

{

cout<<"CUENTA ACTIVA. "<<endl;

}

else

{

cout<<"CUENTA INACTIVA. "<<endl;

}

cout<<"-------------------------------"<<endl;

}

cout<<endl;

cout<<"Presione cualquier tecla para continuar..."<<endl;

getch();

system("cls");

}

void mostrarlote()// LISTAR LOTES POR TARJETAID O LOTES CREADOS DIARIOS

{

FILE\*movimientosAux;

Moviento auxiliar;

int cont=0;

if(movimientosAux=fopen(ARCHIVOLOTESPROCESADOS,"rb"))

{

fread(&auxiliar,sizeof(Moviento),1,movimientosAux);

int minutos=0;

int hora=0;

char opcion4=0;

cout<< "\*\*\*\*\*\*\*\*\*\*\* MOVIMIENTO(S) \*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\* DIGITE UNA OPCION \*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<endl;

cout<<" 1- MOSTRAR MOVIMIENTO POR TARJETA ID. "<<endl;

cout<<" 2- LISTAR TODOS LOS MOVIMIENTOS DIARIOS. "<<endl;

do

{

opcion4=getch();

}while(opcion4!='1' && opcion4!='2');

system("cls");

if(opcion4=='2')

{

while(!feof(movimientosAux))

{

if(opcion4=='2')

{

cout<< "\*\*\*\*\*\*\*\*\*\*\* MOVIMIENTO(S) \*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<endl;

opcion4=0;

}

hora = auxiliar.Hora/100;

minutos = auxiliar.Hora%100;

int anio=double(auxiliar.fecha/10000);

int dia=auxiliar.fecha%100;

int mes=auxiliar.fecha%10000;

int mes1=mes/100;

cout <<" MOVIMIENTO ID :"<<auxiliar.movientoID<<"."<<endl;

cout <<" TARJETA ID :"<<auxiliar.tarjetaID<<"."<<endl;

cout <<" MONTO $ "<<auxiliar.monto<<"."<<endl;

cout <<" FECHA/HORA :"<<anio<<"/"<<mes1<<"/"<<dia<<" "<<hora<<":"<<minutos<<"."<<endl;

cout<<endl<<endl;

cont=1;

fread(&auxiliar,sizeof(Moviento),1,movimientosAux);

}

}

else

{

//MUESTRO MOVIMIENTOS ASOCIADO A UNA TARJETA

cout<< "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* MOVIMIENTO(S) \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<endl;

int tarID=0;

cout<<" INGRESE LA TARJETAID PARA MOSTRAR SUS MOVIMIENTOS DIARIOS "<<endl;

cout<<endl;

cout<<"TARJETAID:" <<endl;

cin>>tarID;

system("cls");

while(!feof(movimientosAux))

{

if(opcion4=='1')

{

cout<< "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* MOVIMIENTO(S) \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<endl;

opcion4=0;

}

if(tarID==auxiliar.tarjetaID)

{

hora = auxiliar.Hora/100;

minutos = auxiliar.Hora%100;

int anio=double(auxiliar.fecha/10000);

int dia=auxiliar.fecha%100;

int mes=auxiliar.fecha%10000;

int mes1=mes/100;

cout <<" MOVIMIENTO ID :"<<auxiliar.movientoID<<"."<<endl;

cout <<" TARJETA ID :"<<auxiliar.tarjetaID<<"."<<endl;

cout <<" MONTO $ "<<auxiliar.monto<<"."<<endl;

cout <<" FECHA/HORA :"<<anio<<"/"<<mes1<<"/"<<dia<<" "<<hora<<":"<<minutos<<"."<<endl;

cout<<endl<<endl;

cont=1;

fread(&auxiliar,sizeof(Moviento),1,movimientosAux);

}

else

{

fread(&auxiliar,sizeof(Moviento),1,movimientosAux);

}

}

}

fclose(movimientosAux);

if( cont==0 ) //VERIFICO SI TENGO ALGUN MOVIMIENTO

{

system("cls");

cout<< "\*\*\*\*\*\*\*\*\*\*\* MOVIMIENTO(S) \*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<endl;

cout<<" NO SE REGISTRA NINGUN MOVIMIENTO EN EL SISTEMA! "<<endl;

cout<<endl;

}

}

else

{

cout<< "\*\*\*\*\*\*\*\*\*\*\* MOVIMIENTO(S) \*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<endl;

cout<<" NO SE REGISTRA NINGUN MOVIMIENTO EN EL SISTEMA! "<<endl;

cout<<endl;

}

system("pause");

}

int desactivarCuenta() //DESACTIVO CUENTAS ACTIVA / INACTIVA

{

FILE\*cuentasBancariasCarga;

Tarjeta auxiliar;

int codigo=0,clienteNro=0;

int cuentaEncontrada=0;

cuentasBancariasCarga=fopen(ARCHIVOCUENTAS,"rb+");

if(cuentasBancariasCarga==NULL)

{

cout<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\* DESACTIVAR TARJETA \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* "<<endl;

cout<<endl;

cout<<" NO FIGURA NINGUNA TARJETA EN EL SISTEMA! "<<endl;

cout<<endl;

system("pause");

return 1;

}

cout<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\* DESACTIVAR TARJETA \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* "<<endl;

cout<<endl;

cout<<"\*\*\*\* INGRESE TARJETAID Y NUMERO DEL CLIENTE A SER DESACTIVADA \*\*\*\*"<<endl;

cout<<endl;

cout<<" TARJETA ID 99999:"<<endl;

cin>>codigo;

cout<<" NUMERO DEL CLIENTE 9999: "<<endl;

cin>>clienteNro;

while(!cuentaEncontrada && fread(&auxiliar,sizeof(Tarjeta),1,cuentasBancariasCarga))

{

//BORRADO DE CUENTA

if(codigo==auxiliar.TarjetaID && clienteNro==auxiliar.NroCliente)

{

cuentaEncontrada=1;

auxiliar.Activa=0;

fseek(cuentasBancariasCarga,(-1)\*sizeof(Tarjeta),SEEK\_CUR);

fwrite(&auxiliar,sizeof(Tarjeta),1,cuentasBancariasCarga);

}

}

fclose(cuentasBancariasCarga);

if(cuentaEncontrada==0)

{

cout<<endl;

cout<<" TARJETA INEXISTENTE EN EL SISTEMA! "<<endl;

cout<<endl;

system("pause");

}

else

{

cout<<endl;

cout<<" TARJETA DESACTIVADA EXITOSAMENTE! "<<endl;

cout<<endl;

cout<<"Presione cualquier tecla para continuar..."<<endl;

getch();

}

}

bool cargar\_cuenta\_bancaria() //CREO NUEVAS CUENTAS

{

FILE\*cuentasBancariasCarga;

Tarjeta vc;

cout<<" \*\*\*\*\*\*\*\* INGRESE LOS SIGUIENTES DATOS \*\*\*\*\*\*\* "<<endl;

cout<<endl;

cout<<" ID TARJETA 99999 "<<endl;

cin>>vc.TarjetaID;

if(!buscarTarjeta(vc.TarjetaID,4))

{

vc.Saldo=0;

cout<<" NUMERO DEL CLIENTE 9999:"<<endl;

cin>>vc.NroCliente;

cout<<" FECHA DE CREACION AAAAMMDD"<<endl;

cin>>vc.fechaCreacion;

vc.Activa=1;

}

else

{

return false;

}

cuentasBancariasCarga=fopen(ARCHIVOCUENTAS,"ab");

if(cuentasBancariasCarga==NULL)

{

cout<<" ERROR AL ABRIR CUENTAS! "<<endl;

return false;

}

else

{

fwrite(&vc,sizeof(Tarjeta),1,cuentasBancariasCarga);

fclose(cuentasBancariasCarga);

return true;

}

}

int main()

{

char salida;

mkdir("abc");

do

{

salida = menucuentas();

}while (salida != '6');

cout<<endl;

cout<< " GRACIAS! "<<endl;

cout<<endl;

cout<<" \*\*\*\*\*\*\* FIN \*\*\*\*\*\*\* "<<endl;

return 0;

}