#include <stdio.h>

#include <string.h>

#include <conio.h>

#include <stdlib.h>

typedef struct contaBancaria{

char Nconta[50],nome[100],cpf[13],fone[15];

float saldo;

}conta;

int buscar(conta vet[],char nuncont[],int qtd);//

void cadast(conta vet[],char nuncont[],int \*qtd);

void consulta(conta vet[],char nuncont[],int qtd);//

void depositar(conta vet[],char nuncont[],int qtd);//

void sacar(conta vet[],char nuncont[],int qtd);//

void exibir(conta vet[],int qtd,conta aux[]);

void qtdCPF(conta vetC[],conta vetP[],char CPF[]);//

void remover(conta vet[],char nuncont[],int \*qtd);

void zerar(conta vet[]);

int main(){

conta corrente[100],poupan[100],aux[2];

int qtdC=0,qtdP=0;

char resp,nunNovaConta[50],CPFaux[13];

zerar(corrente);

zerar(poupan);

do{

printf("Qual o opcao voce deseja :\n1-Cadastro\n2-Consultar Saldo\n3-Deposito\n4-Saque\n5-Exibir\n6-Remover conta\n");

printf("7-Quantidade de Contas\n8-Sair\n");

resp=getchar();fflush(stdin);

system("cls");

switch(resp){

case '1' :

printf("Digite o numero da nova conta :");

gets(nunNovaConta);fflush(stdin);

printf("\nQue tipo de conta voce deseja cadastrar:\n1 - Corrente\n2 - Poupanca\n");

resp = getchar();fflush(stdin);

if(resp == '1'){

cadast(corrente,nunNovaConta,&qtdC);

break;

}

else{

cadast(poupan,nunNovaConta,&qtdP);

break;

}

case '2' :

printf("Qual o tipo de conta vai ser consultada:\n1 - Corrente\n2 - Poupanca\n");

resp=getchar();fflush(stdin);

printf("Digite o numero da conta que sera consultada:");

gets(nunNovaConta);fflush(stdin);

if(resp = '1'){

consulta(corrente,nunNovaConta,qtdC);

break;

}

else{

consulta(poupan,nunNovaConta,qtdP);

break;

}

case '3' :

printf("Qual o tipo de conta vai ser consultada:\n1 - Corrente\n2 - Poupanca\n");

resp=getchar();fflush(stdin);

printf("Digite o numero da conta que sera consultada:");

gets(nunNovaConta);fflush(stdin);

if(resp == '1'){

depositar(corrente,nunNovaConta,qtdC);

break;

}

else{

depositar(poupan,nunNovaConta,qtdP);

break;

}

break;

case '4' :

printf("Qual o tipo de conta, que sera sacada:\n1 - Corrente\n2 - Poupanca\n");

resp = getchar();fflush(stdin);

printf("Digite o numero da conta: ");

gets(nunNovaConta);

if(resp == '1'){

sacar(corrente,nunNovaConta,qtdC);

break;

}

else{

sacar(poupan,nunNovaConta,qtdP);

break;

}

case '5' :

printf("Escolha o tipo das contas:\n1 - Corrente\n2 - Poupanca\n");

resp = getchar();fflush(stdin);

if(resp == '1'){

exibir(corrente,qtdC,aux);

break;

}

else{

exibir(poupan,qtdP,aux);

break;

}

case '6' :

printf("Qual o tipo de conta:\n1 - Corrente\n2 - Poupanca\n");

resp = getchar();fflush(stdin);

printf("Digite o numero da conta: ");

gets(nunNovaConta);fflush(stdin);

if(resp == '1'){

remover(corrente,nunNovaConta,&qtdC);

break;

}

else{

remover(poupan,nunNovaConta,&qtdP);

break;

}

case '7' :

printf("Digite o CPF: ");

fgets(CPFaux,12,stdin);fflush(stdin);

qtdCPF(corrente,poupan,CPFaux);

break;

}

}while(resp != '8'&&(qtdC < 100 && qtdP < 100));

return 0;

}

void zerar(conta vet[]){

int i;

char aux[100]=" ";

for(i=0;i<100;i++){

strcpy(vet[i].Nconta,aux);

strcpy(vet[i].cpf,aux);

strcpy(vet[i].nome,aux);

strcpy(vet[i].fone,aux);

}

}

int buscar(conta vet[],char nuncont[],int qtd){

int i,resp;

char aux[50];

strcpy(aux,nuncont);

if(qtd == 0)

return -2;

else{

for(i=0;i<100;i++){

resp=strcmp(aux,vet[i].Nconta);

if(resp == 0)

return i;

}

return -1;

}

}

void cadast(conta vet[],char nuncont[],int \*qtd){

char resp;

resp = buscar(vet,nuncont,\*qtd);

if(resp == -2)

printf("Nao ha contas registradas\n");

if(resp != -1 && resp != -2)

printf("Voce nao pode realizar o cadastro, pois o numero da conta ja existe!\n");

else{

strcpy((vet[\*qtd].Nconta),nuncont);fflush(stdin);

printf("Digite seu CPF : ");

fgets(vet[\*qtd].cpf,12,stdin);fflush(stdin);

printf("\nDigite seu nome:");

gets(vet[\*qtd].nome);fflush(stdin);

printf("\nDigite telefone:");

fgets(vet[\*qtd].fone,14,stdin);fflush(stdin);

printf("\nDigite seu saldo:");

scanf("%f",&(vet[\*qtd].saldo));fflush(stdin);

(\*qtd)++;

}

}

void consulta(conta vet[],char nuncont[],int qtd){

char resp;

resp=buscar(vet,nuncont,qtd);

if(resp == -2)

printf("Nao ha contas registradas\n");

else if(resp == -1)

printf("Conta nao cadastrada!\n");

else{

printf("O saldo da sua conta e : %.2f\n",vet[resp].saldo);

}

}

void depositar(conta vet[],char nuncont[],int qtd){

float valor;

char resp;

resp=buscar(vet,nuncont,qtd);

if(resp == -2)

printf("Nao ha contas registradas!\n");

else if(resp == -1)

printf("Conta nao cadastrada\n");

else{

printf("Digite o valor a ser depositado:");

scanf("%f",&valor);

vet[resp].saldo= (vet[resp].saldo) + valor;

}

}

void sacar(conta vet[],char nuncont[],int qtd){

float valor;

char resp;

resp=buscar(vet,nuncont,qtd);

if(resp == -2)

printf("Nao ha contas registradas\n");

else if(resp == -1)

printf("Conta nao cadastrada!\n");

else{

printf("Digite quanto deseja sacar:");

scanf("%f",&valor);

if(vet[resp].saldo<valor)

printf("Saldo insuficiente!\n");

else

vet[resp].saldo = vet[resp].saldo - valor;

}

}

void exibir(conta vet[],int qtd,conta aux[]){

int i,resp,x=0;

if(qtd>0){

while(x<=qtd){

for(i=0;i<qtd;i++){

resp = strcmp(vet[i].nome,vet[i+1].nome);

if(resp > 0){

aux[1]=vet[i];

vet[i] = vet[i+1];

vet[i+1]= aux[1];

}

}

x++;

}

for(i=0;i<qtd;i++){

printf("Numero da conta: %s",vet[i].Nconta);

printf("\nNome do titular: %s",vet[i].nome);

printf("\ntelefone para contato: %s\n",vet[i].fone);

}

}

else

printf("Nao ha contas registradas!\n\n");

}

void remover(conta vet[],char nuncont[],int \*qtd){

int i;

char resp;

resp=buscar(vet,nuncont,\*qtd);

if(resp == -2)

printf("Nao ha contas registradas\n\n");

else if(resp == -1)

printf("Conta nao cadastrada!\n");

else{

if(vet[resp].saldo!=0)

printf("A conta nao pode ser removida, Pois seu saldo e de : %f",vet[resp].saldo);

else{

for(i=resp;i<=(\*qtd);i++)

vet[i] = vet[i+1];

\*qtd--;

}

}

}

void qtdCPF(conta vetC[],conta vetP[],char CPF[],int qtd){

int i,resp;

for(i=0;i<100;i++){

resp=strcmp((vetC[i].cpf),CPF);

if(resp==0)

printf("Numero da conta: %s \nTipo da conta:(Corrente)\n",vetC[i].Nconta);

resp=strcmp((vetP[i].cpf),CPF);

if(resp==0)

printf("Numero da conta: %s \nTipo da conta:(Poupanca)\n",vetP[i].Nconta);

}

}