FUNCS.C:

#include <string.h>

#include <stdlib.h>

#include "menu.h"

#include "funcs.h"

ACCOUNT \*addAccount(ACCOUNT \*array,short \*memoryFlag,short \*numberElements,short submenu,short submenu1,short\*lenghtArray)

{

if(\*memoryFlag==0)

{

array=(ACCOUNT\*)malloc(rows\*sizeof(ACCOUNT));

if(array==NULL)

{

printf("\nNe e sazdaden spisak!");

exit(1);

}

\*memoryFlag=1;

\*lenghtArray=rows;

}

array=submenu==1?addAccountFromKey(array,\*memoryFlag,numberElements,submenu1,lenghtArray):addAccountFromFile(array,\*memoryFlag,numberElements,submenu1,lenghtArray);

return array;

}

---------------------------------------------------------------

ACCOUNT \*addAccountFromKey(ACCOUNT \*array,short memoryFlag,short \*numberElements,short submenu1,short\*lenghtArray)

{

ACCOUNT a;

char word[4];

char kod1[20];

if(submenu1==1)

{

\*numberElements=0;

}

while(printf("Shte vyvedete li nov klient? - (DA/NE)"),strcmp(gets(word),"da")==0)

{

if(\*numberElements<\*lenghtArray)

{

printf("\nKLIENT NOMER %d: ",\*numberElements+1);

inputAccount(&a);

strcpy(kod1,a.kod);

if(isKod(kod1,array,numberElements)==0)

memcpy(&array[\*numberElements],&a,sizeof(ACCOUNT));

else{

printf("Sashtestvyva klient s takav kod!\n");

(\*numberElements)--;

}

}

else

{

\*lenghtArray=\*lenghtArray\*2;

array=(ACCOUNT\*)realloc(array,\*lenghtArray\*sizeof(ACCOUNT));

memcpy(&array[\*numberElements],&a,sizeof(ACCOUNT));

}

(\*numberElements)++;

fflush(stdin);

}

return array;

}

//////////////////////////////////////////////////////////////////////////////////////////

ACCOUNT \*addAccountFromFile(ACCOUNT \*array,short memoryFlag,short \*numberElements,short submenu1,short\*lenghtArray)

{

FILE \*fp\_out=NULL;

char fname[256];

short n=\*lenghtArray;;

if(memoryFlag==0)

{

printf("\nNe e syzdaden spisak!");

exit(1);

}

fflush(stdin);

if(submenu1==1)

{

\*numberElements=0;

}

puts("Vyvedete imeto na faila ot koito shte se chete");

fflush(stdin);

gets(fname);

fp\_out=fopen(fname,"r+b");

if(fp\_out==NULL)

{

printf( "Failat ne moje da bade otvoren\n" );

exit(1);

}

while(1)

{

if(\*numberElements<\*lenghtArray)

{

if(fread(&array[\*numberElements],sizeof(ACCOUNT),1,fp\_out)!=1)

{

if(ferror(fp\_out))

{

perror("Gresha pri cheteneto :");

break;

}

if(feof(fp\_out))

break;

}

(\*numberElements)++;

}

else

{

\*lenghtArray=n=\*lenghtArray\*2;

array=(ACCOUNT\*)realloc(array,\*lenghtArray\*sizeof(ACCOUNT));

if(fread(&array[\*numberElements],sizeof(ACCOUNT),1,fp\_out)!=1)

{

if(ferror(fp\_out))

{

perror("Gresha pri cheteneto :");

break;

}

if(feof(fp\_out))

break;

}

(\*numberElements)++;

}

}

fclose(fp\_out);

return array;

}

//////////////////////////////////////////////////////////////////////////////////////////

void inputAccount(ACCOUNT \*a)

{

printf("\nVyvedete koda na smetkata :");

fflush(stdin);

scanf("%s",&a->kod);

printf("\nVyvedete imeto na titulqra:");

fflush(stdin);

gets(a->name);

printf("\nVyvedete sumata po smetkata:");

fflush(stdin);

scanf("%f",&a->sum);

printf("\nVyvedete broq na mesecite prez koito e syshtesvyvala smetkata :");

fflush(stdin);

scanf("%d",&a->months);

}

//////////////////////////////////////////////////////////////////////////////////////////

void printArray(ACCOUNT \*array,short n)

{

short i;

if(n>0)

for(i=0;i<n;i++)

printAccounts(array,i);

else

printf("\nSpisaka sys klietni e prazen!\n");

}

/////////////////////////////////////////////////////////////////////////////////////////////////

void printAccounts(ACCOUNT \*array,short i)

{

printf("\nNomer na smetkata %d",i+1);

printf("\nKod na smetkata :%s",array[i].kod);

printf("\nIme na titulqra :%s",array[i].name);

printf("\nSuma po smetkata :%.4f",array[i].sum);

printf("\nMeseci prestoi v bankata :%d\n",array[i].months);

}

//////////////////////////////////////////////////////////////////////////////////////////////////

void writeStructFile(ACCOUNT \*array,short numberElements)

{

FILE \*fp\_in=NULL;

char fname[256];

short i;

if(numberElements>0)

{

puts("Vyvedete ime na faila za zapis");

fflush(stdin);

if(gets(fname)==NULL)

printf("Greshka v imeto na faila");

fp\_in=fopen(fname,"w+b");

if(fp\_in==NULL)

{

printf( "Problem s otvarqneto na faila\n" );

exit(1);

}

for(i=0;i<numberElements;i++)

{

fwrite(&array[i],sizeof(ACCOUNT),1,fp\_in);

if(ferror(fp\_in))

{

perror("Greshka ri zapisvaneto :");

break;

}

}

fclose(fp\_in);

}

else

printf("\nSpisakyt s klienti e prazen!\n");

system("pause");

}

/////////////////////////////////////////////////////////////////////////////////////////

ACCOUNT \*deleteArray(ACCOUNT \*array,short \*n,short \*p)

{

if(array!=NULL)

{

free(array);

\*n=0;

\*p=0;

printf("\nSpisykyt beshe iztrit!");

}

else

printf("\nNqma nalichen spisak!");

return array;

}

///////////////////////////////////////////////////////////////////////////////

void findByKod(ACCOUNT \*array,short n)

{

short i,flag=0;

char kod[20];

puts("Vyvedete koda na smetkata za tyrsene :");

fflush(stdin);

if(gets(kod)==NULL)

printf("Greshka pri vyvejdane");

for(i=0;i<n;i++)

{

if(strcmp(array[i].kod,kod)==0)

{

printf("\nNameren klient s poreden nomer:");

printAccounts(array,i);

flag++;

}

}

if(flag==0)

printf("\nKlient s takyv kod ne sychtestvyva!");

}

///////////////////////////////////////////////////////////////////////////////

void deleteElement(ACCOUNT \*array,short n,short j)

{

array[j]=array[n-1];

}

//////////////////////////////////////////////////////////////////////////////////////////

short deleteElementByKod(ACCOUNT \*array,short n)

{

short i,flag=0;

char kod[20];

puts("Vyvedete koda na klienta za iztrivane :");

fflush(stdin);

if(gets(kod)==NULL)

printf("Greshka pri vyvejdaneto");

for(i=0;i<n;i++)

{

if(strcmp(array[i].kod,kod)==0)

{

deleteElement(array,n,i);

flag++;

}

}

if(flag!=0)

n=n-1;

else

printf("\nKlient s takyv kod ne sashtestvyva!");

return n;

}

/////////////////////////////////////////////////////////////////////////////////////////////

float findavgVal(ACCOUNT \*array,short n)

{

short i,flag=0;

float sum=0;

for(i=0;i<n;i++)

{

sum+=array[i].sum;

}

return (sum/n);

}

void pravgArray(ACCOUNT \*array,short n)

{

short i;

if(n>0)

for(i=0;i<n;i++)

{

if(array[i].sum>findavgVal(array,n))

prAccountsavg(array,i);

}

else

printf("\nSpisaka sys klietni e prazen!\n");

system("pause");

}

void prAccountsavg(ACCOUNT \*array,short i)

{

printf("\nKod na smetkata :%s",array[i].kod);

printf("\nIme na titulqra :%s",array[i].name);

}

int isKod(char \*a,ACCOUNT \*array,short n)

{

int is=1;

short i;

for( i=0;i<n;i++)

{

if(strcmp(array[i].kod,a)==0)is=0;

}

return is;

}

FUNCS.H:

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <memory.h>

typedef struct

{

char kod[20];

char name[50];

float sum;

int months;

//struct dateOfBirth date\_a;

}ACCOUNT;

//--------------------------------------------

ACCOUNT \*addAccount(ACCOUNT \*array,short \*memoryFlag,short \*countFlag,short submenu,short submenu1,short\*lenghtArray);

ACCOUNT \*addAccountFromKey(ACCOUNT \*array,short memoryFlag,short \*i,short,short\*lenghtArray);

ACCOUNT \*addAccountFromFile(ACCOUNT \*array,short memoryFlag,short \*i,short,short\*lenghtArray);

void inputAccount(ACCOUNT\*);

void printArray(ACCOUNT \*,short );

void printAccounts(ACCOUNT \*array,short i);

void writeStructFile(ACCOUNT \*,short);

ACCOUNT \*deleteArray(ACCOUNT \*array,short \*n,short \*p);

void findBykod(ACCOUNT \*array,short n);

void deleteElement(ACCOUNT \*array,short n,short j);

short deleteElementBykod(ACCOUNT \*array,short n);

float avgVal(ACCOUNT \*array,short n);

void prAccountsavg(ACCOUNT \*array,short i);

void pravgArray(ACCOUNT \*array,short n);

int isKod(char \*a,ACCOUNT \*array,short n);

// MENU.h

short menu( char \*t[]);

// MENU . c

#include <stdio.h>

#include <conio.h>

#include <stdlib.h>

short menu( char \*t[])

{

int k,i=1;

char ch;

system("cls");

printf ("\n%s\n",t[0]);

while (t[i])

printf("\n%d.%s", i-1,t[i++]);

printf ("\n\nIzberete operaciq:");

if(i<8)

i--;

do

{

ch=getch();

k=ch-'0';

} while(k<=0 || k >i);

printf("%c\n",ch);

return ch-'0';

}

//MAIN.C

#include <stdio.h>

#include <stdlib.h>

#include <windows.h>

#include "menu.h"

#include "funcs.h"

char \*menu\_main[]=

{

"Program <Bank>\n--------------------------",

"Dobavqne na nov klient",

"Zapis na spisaka s klientite vyv fail",

"Tarsene na klient po kod",

"Iztrivane na klient po kod",

"Pokazvane na sipisaka s klienti",

"Spisak s klienti sys suma po golqma ot srednata" ,

"Iztrivane na tekushtiq spisak s klienti",

"Krai",

NULL

};

char \*menu\_1[]=

{

"---VHOD---",

"Ot klaviaturata",

"Ot fail",

NULL

};

char \*menu\_1\_1[]=

{

"IZBOR NA MQSTO ZA DOBAVQNE",

"Nov spisak s klienti",

"Kam sashtestvuvashtiya spisak",

NULL

};

int main()

{

short mode, submenu,submenu1,memoryFlag=0,numberElements=0,sortFlag=0,lenghtArray=0;

ACCOUNT \*array=NULL;

//------------------------------------------------------

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

//---------------------------------------

do

{

mode=menu(menu\_main);

switch (mode)

{

case 1: submenu=menu(menu\_1);

submenu1=menu(menu\_1\_1);

array=addAccount(array,&memoryFlag,&numberElements,submenu,submenu1,&lenghtArray);break;

case 2:writeStructFile(array,numberElements);break;

case 3:if(numberElements)

{

findByKod(array,numberElements);system("pause");break;

}

else

{ printf("\nSpisaka sys klietni e prazen!\n");system("pause");}

break;

case 4:if(numberElements)

{

numberElements=deleteElementByKod(array,numberElements);system("pause");break;

}

else

{ printf("\nSpisaka sys klietni e prazen!\n");system("pause");}

break;

case 5:printArray(array,numberElements);system("pause");break;

case 7:array=deleteArray(array,&numberElements , &lenghtArray);system("pause");

case 6:{if(numberElements){pravgArray(array,numberElements);break;}else{printf("Spisakat s klienti e prazen");system("pause");}break;}

}

}while (mode != 8);

array=deleteArray(array,&numberElements, &lenghtArray);

}