#include <stdio.h>

#include <conio.h>

#include <windows.h>

#include <string.h>

#include <stdlib.h>

typedef struct{

char number[11];

char name[81];

char izdatel[51];

int nali4nost;

int year;

int mount;

int day;

char avtorime[21];

char avtorprezime[21];

char avtorfamiliq[21];

}BOOK;

typedef struct list{

BOOK kniga;

struct list \*next;

}list;

int broi=0;

list\* addbook(list\* prev)

{

list\* spisak;

char number[11];

char name[81];

char izdatel[51];

int nali4nost;

int year;

int mount;

int day;

char avtorime[21];

char avtorprezime[21];

char avtorfamiliq[21];

int check,i;

spisak=(list \*)malloc(sizeof(list));

spisak->next=prev;

prev=spisak;

for(spisak=prev;;spisak=prev)

{

printf("Въведете библиотечен номер: ");

fflush(stdin);

scanf("%s",number);

if( number[0]>'Я' || number[0]<'А' || number[3]!='-' ||

number[1]>'9' || number[1]<'0' || number[2]>'9' || number[2]<'0' ||

number[4]>'9' || number[4]<'0' || number[5]>'9' || number[5]<'0' ||

number[6]>'Я' || number[6]<'А' || number[7]>'9' || number[7]<'0' ||

number[8]>'Я' || number[8]<'А' || number[9]>'Я' || number[9]<'А')

{

continue;

}

for(;;spisak=spisak->next)

{

if(strcmp(spisak->kniga.number,number)==0)

{

break;

}

if(spisak->next==NULL)

{

break;

}

}

if(strcmp(spisak->kniga.number,number)==0)

{

printf("\n\nВече има въведена книга с такъв библиотечен номер!\n\n ");

continue;

}

else

{

break;

}

}

spisak=prev;

strcpy(spisak->kniga.number,number);

do{

check=0;

printf("Въведете име на книгата: ");

fflush(stdin);

scanf("%s",name);

for(i=0;i<(strlen(name)-1);i++)

{

if( (!(name[i]>='a'&&name[i]<='z')) && (!(name[i]>='A'&&name[i]<='Z')) &&

(!(name[i]>='0'&&name[i]<='9')) && (!(name[i]>='а'&&name[i]<='я')) &&

(!(name[i]>='А'&&name[i]<='Я')) )

{

check=1;

break;

}

}

if(check)

{

check=0;

continue;

}

if( (name[0]>='A' && name[0]<='Z') || (name[0]>='А' && name[0]<='Я') )

{

break;

}

}while(1);

strcpy(spisak->kniga.name,name);

printf("Въведете име на издателството: ");

fflush(stdin);

fgets(izdatel,51,stdin);

strcpy(spisak->kniga.izdatel,izdatel);

do{

printf("Въведете наличност(y,n,b,z): ");

fflush(stdin);

nali4nost=getchar();

}while(nali4nost!='y' && nali4nost!='n' && nali4nost!='b' && nali4nost!='z');

spisak->kniga.nali4nost=nali4nost;

do{

printf("Въведете година на издаване: ");

fflush(stdin);

scanf("%d",&year);

if(year>2013 || year<1)

{

printf("Въведете година между 1 и 2013.\n");

continue;

}

else

{

break;

}

}while(1);

do{

printf("Въведете месец на издаване: ");

fflush(stdin);

scanf("%d",&mount);

if(mount<1 || mount>12)

{

printf("Невалидна дата\n");

continue;

}

else

{

break;

}

}while(1);

do{

printf("Въведете ден на издаване: ");

fflush(stdin);

scanf("%d",&day);

if(mount==1||mount==3||mount==5||mount==7||mount==8||mount==10||mount==12)

{

if(day>31||day<1)

{

printf("Невалидна дата\n");

continue;

}

else

{

break;

}

}

if(mount==4||mount==6||mount==9||mount==11)

{

if(day>30||day<1)

{

printf("Невалидна дата\n");

continue;

}

else

{

break;

}

}

if(mount==2)

{

if(year%4==0)

{

if(day>29||day<1)

{

printf("Невалидна дата\n");

continue;

}

else

{

break;

}

}

else

{

if(day>28||day<1)

{

printf("Невалидна дата\n");

continue;

}

else

{

break;

}

}

}

}while(1);

spisak->kniga.year=year;

spisak->kniga.day=day;

spisak->kniga.mount=mount;

printf("Въведете име на автора: ");

fflush(stdin);

scanf("%s",avtorime);

strcpy(spisak->kniga.avtorime,avtorime);

printf("Въведете презиме на автора: ");

fflush(stdin);

scanf("%s",avtorprezime);

strcpy(spisak->kniga.avtorprezime,avtorprezime);

printf("Въведете фамилия на автора: ");

fflush(stdin);

scanf("%s",avtorfamiliq);

strcpy(spisak->kniga.avtorfamiliq,avtorfamiliq);

broi++;

return prev;

}

list\* dellbook(list\* prev)

{

list\* spisak, \*head2;

char number[11];

int check;

for(check=0;;)

{

printf("Въведете библиотечен номер: ");

fflush(stdin);

scanf("%s",number);

if( number[0]>'Я' || number[0]<'А' || number[3]!='-'||

number[1]>'9' || number[1]<'0' || number[2]>'9' || number[2]<'0' ||

number[4]>'9' || number[4]<'0' || number[5]>'9' || number[5]<'0' ||

number[6]>'Я' || number[6]<'А' || number[7]>'9' || number[7]<'0' ||

number[8]>'Я' || number[8]<'А' || number[9]>'Я' || number[9]<'А')

{

printf("Въвели сте невалиден номер!\n\n");

continue;

}

for(head2=NULL,spisak=prev;spisak!=NULL;head2=spisak,spisak=spisak->next)

{

if(strcmp(spisak->kniga.number,number)==0)

{

if(head2!=NULL)

{

head2->next=spisak->next;

}

else

{

prev=spisak->next;

}

check=1;

broi--;

printf("\nУспешно изтрита!\n\n");

free(spisak);

break;

}

if(spisak->next==NULL)

{

break;

}

}

if(!check)

{

printf("\nНе беше намерена книга с такъв номер.\n\n");

}

break;

}

return prev;

}

list\* printone(list\* prev)

{

list\* spisak;

int check;

char number[11];

for(check=0;;spisak=prev)

{

printf("Въведете библиотечен номер: ");

fflush(stdin);

scanf("%s",number);

if( number[0]>'Я' || number[0]<'А' || number[3]!='-'||

number[1]>'9' || number[1]<'0' || number[2]>'9' || number[2]<'0' ||

number[4]>'9' || number[4]<'0' || number[5]>'9' || number[5]<'0' ||

number[6]>'Я' || number[6]<'А' || number[7]>'9' || number[7]<'0' ||

number[8]>'Я' || number[8]<'А' || number[9]>'Я' || number[9]<'А')

{

printf("Въвели сте невалиден номер!\n\n");

continue;

}

for(spisak=prev;spisak!=NULL;spisak=spisak->next)

{

if(strcmp(spisak->kniga.number,number)==0)

{

printf("Библиотечен номер: %s",spisak->kniga.number);

printf("\nИме: %s",spisak->kniga.name);

printf("Издателство: %s",spisak->kniga.izdatel);

if(spisak->kniga.nali4nost=='y')

{

printf("Наличност: Взета");

}else if(spisak->kniga.nali4nost=='n')

{

printf("Наличност: Налична");

}else if(spisak->kniga.nali4nost=='b')

{

printf("Наличност: Бракувана");

}else if(spisak->kniga.nali4nost=='z')

{

printf("Наличност: Запазена");

}

printf("\nДата на издаване: %d.%d.%d",spisak->kniga.day,spisak->kniga.mount,spisak->kniga.year);

printf("\nАвтор: %s %s %s",spisak->kniga.avtorime,spisak->kniga.avtorprezime,spisak->kniga.avtorfamiliq);

check=1;

break;

}

if(spisak->next==NULL)

{

break;

}

}

if(!check)

{

printf("\nНе беше намерена книга с такъв номер.\n\n");

}

break;

}

return prev;

}

list\* zapis(list\* prev)

{

int check,i;

list \*spisak;

FILE \*fout;

char foutname[256];

for(check=0;;)

{

printf("Въведете файл за запис: ");

fflush(stdin);

scanf("%s",foutname);

i=strlen(foutname);

if(foutname[i-1]=='t'&&foutname[i-2]=='x'&&foutname[i-3]=='t')

{

fout=fopen(foutname,"w");

if(fout!=NULL)

{

for(i=0,spisak=prev;;spisak=spisak->next,i++)

{

if(i!=0)fprintf(fout,"\n");

fprintf(fout,"%s",spisak->kniga.number);

fprintf(fout,"\n%s\n",spisak->kniga.name);

fprintf(fout,"%s\n",spisak->kniga.izdatel);

fprintf(fout,"%c\n",spisak->kniga.nali4nost);

fprintf(fout,"%d.%d.%d\n",spisak->kniga.day,spisak->kniga.mount,spisak->kniga.year);

fprintf(fout,"%s\n%s\n%s\n",spisak->kniga.avtorime,spisak->kniga.avtorprezime,spisak->kniga.avtorfamiliq);

check=1;

if(spisak->next==NULL)break;

}

}

else

{

printf("\nФайла не беше отворен.");

}

break;

}

else

{

printf("\nГрешка при отваряне на файла.Трябва да е от тип .txt\n");

}

}

if(fout!=NULL) fclose(fout);

return prev;

}

list\* readfile(list\* prev)

{

int check,i,ch,cn;

list \*spisak, \*head;

FILE \*fin;

char finname[256];

for(broi=0,check=0;;)

{

printf("Въведете файл за четене: ");

fflush(stdin);

scanf("%s",finname);

i=strlen(finname);

if(finname[i-1]=='t'&&finname[i-2]=='x'&&finname[i-3]=='t')

{

fin=fopen(finname,"r");

if(fin!=NULL)

{

if(prev!=NULL)

{

for(spisak=prev;;spisak=head)

{

head=spisak->next;

free(spisak);

if(head==NULL) break;

}

}

prev=NULL;

do{

spisak=(list \*)malloc(sizeof(list));

spisak->next=prev;

prev=spisak;

for(i=0;;i++)

{

ch=fgetc(fin);

if(ch!='\n')

{

spisak->kniga.number[i]=ch;

}

else

{

spisak->kniga.number[i]='\0';

break;

}

}

for(i=0;;i++)

{

ch=fgetc(fin);

if(ch!='\n')

{

spisak->kniga.name[i]=ch;

}

else

{

spisak->kniga.name[i]='\0';

break;

}

}

for(i=0;;i++)

{

ch=fgetc(fin);

if(ch!='\n')

{

spisak->kniga.izdatel[i]=ch;

}

else

{

spisak->kniga.izdatel[i]='\0';

break;

}

}

ch=fgetc(fin);

spisak->kniga.nali4nost=ch;

ch=fgetc(fin);

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

ch=fgetc(fin);

cn=fgetc(fin);

if(cn=='.') spisak->kniga.day=ch-48;

else

{

spisak->kniga.day=(int)(ch-48)\*10+(cn-48);

ch=fgetc(fin);

}

ch=fgetc(fin);

cn=fgetc(fin);

if(cn=='.') spisak->kniga.mount=(int)(ch-48);

else

{

spisak->kniga.mount=(int)(ch-48)\*10+(cn-48);

ch=fgetc(fin);

}

for(spisak->kniga.year=0,i=1000;;i=i/10)

{

ch=fgetc(fin);

if(ch=='\n')break;

spisak->kniga.year=spisak->kniga.year+(ch-48)\*i;

}

if(i!=0) spisak->kniga.year=spisak->kniga.year/(i\*10);

for(i=0;;i++)

{

ch=fgetc(fin);

if(ch!='\n')

{

spisak->kniga.avtorime[i]=ch;

}

else

{

spisak->kniga.avtorime[i]='\0';

break;

}

}

for(i=0;;i++)

{

ch=fgetc(fin);

if(ch!='\n')

{

spisak->kniga.avtorprezime[i]=ch;

}

else

{

spisak->kniga.avtorprezime[i]='\0';

break;

}

}

for(i=0;;i++)

{

ch=fgetc(fin);

if(ch!=EOF&&ch!='\n')

{

spisak->kniga.avtorfamiliq[i]=ch;

}

else

{

spisak->kniga.avtorfamiliq[i]='\0';

broi++;

break;

}

}

}while(ch=fgetc(fin)!=EOF);

}

else

{

printf("\nФайла не беше отворен.");

}

break;

}

else

{

printf("\nГрешка при отваряне на файла.Трябва да е от тип .txt\n");

}

}

fclose(fin);

return prev;

}

list\* printall(list\* prev)

{

list\* spisak;

int check;

for(check=0,spisak=prev;spisak!=NULL;spisak=spisak->next)

{

printf("Библиотечен номер: %s",spisak->kniga.number);

printf("\nИме: %s",spisak->kniga.name);

printf("\nИздателство: %s\n",spisak->kniga.izdatel);

if(spisak->kniga.nali4nost=='y')

{

printf("Наличност: Взета");

}else if(spisak->kniga.nali4nost=='n')

{

printf("Наличност: Налична");

}else if(spisak->kniga.nali4nost=='b')

{

printf("Наличност: Бракувана");

}else if(spisak->kniga.nali4nost=='z')

{

printf("Наличност: Запазена");

}

printf("\nДата на издаване: %d.%d.%d",spisak->kniga.day,spisak->kniga.mount,spisak->kniga.year);

printf("\nАвтор: %s %s %s\n\n\n\n",spisak->kniga.avtorime,spisak->kniga.avtorprezime,spisak->kniga.avtorfamiliq);

}

return prev;

}

list\* lookfor(list\* prev)

{

int i,check;

char avtorime[21];

char avtorprezime[21];

char avtorfamiliq[21];

list\* spisak;

do{

printf("За търсене по фамилия въведете 1, а за търсене по пълно име въведете 2: ");

fflush(stdin);

scanf("%d",&i);

if(i==1||i==2) break;

}while(1);

if(i==1)

{

printf("Въведете фамилия на автора: ");

fflush(stdin);

scanf("%s",avtorfamiliq);

}

else

{

printf("Въведете име на автора: ");

fflush(stdin);

scanf("%s",avtorime);

printf("Въведете презиме на автора: ");

fflush(stdin);

scanf("%s",avtorprezime);

printf("Въведете фамилия на автора: ");

fflush(stdin);

scanf("%s",avtorfamiliq);

}

switch(i)

{

case 1:

for(spisak=prev;spisak!=NULL;spisak=spisak->next)

{

if(strcmp(spisak->kniga.avtorfamiliq,avtorfamiliq)==0)

{

printf("Библиотечен номер: %s",spisak->kniga.number);

printf("\nИме: %s",spisak->kniga.name);

printf("Издателство: %s",spisak->kniga.izdatel);

if(spisak->kniga.nali4nost=='y')

{

printf("Наличност: Взета");

}else if(spisak->kniga.nali4nost=='n')

{

printf("Наличност: Налична");

}else if(spisak->kniga.nali4nost=='b')

{

printf("Наличност: Бракувана");

}else if(spisak->kniga.nali4nost=='z')

{

printf("Наличност: Запазена");

}

printf("\nДата на издаване: %d.%d.%d",spisak->kniga.day,spisak->kniga.mount,spisak->kniga.year);

printf("\nАвтор: %s %s %s",spisak->kniga.avtorime,spisak->kniga.avtorprezime,spisak->kniga.avtorfamiliq);

}

}

case 2:

for(spisak=prev;spisak!=NULL;spisak=spisak->next)

{

if((strcmp(spisak->kniga.avtorfamiliq,avtorfamiliq)==0) &&

(strcmp(spisak->kniga.avtorime,avtorime)==0) &&

(strcmp(spisak->kniga.avtorprezime,avtorprezime)==0))

{

printf("Библиотечен номер: %s",spisak->kniga.number);

printf("\nИме: %s",spisak->kniga.name);

printf("Издателство: %s",spisak->kniga.izdatel);

if(spisak->kniga.nali4nost=='y')

{

printf("Наличност: Взета");

}else if(spisak->kniga.nali4nost=='n')

{

printf("Наличност: Налична");

}else if(spisak->kniga.nali4nost=='b')

{

printf("Наличност: Бракувана");

}else if(spisak->kniga.nali4nost=='z')

{

printf("Наличност: Запазена");

}

printf("\nДата на издаване: %d.%d.%d",spisak->kniga.day,spisak->kniga.mount,spisak->kniga.year);

printf("\nАвтор: %s %s %s",spisak->kniga.avtorime,spisak->kniga.avtorprezime,spisak->kniga.avtorfamiliq);

check=1;

}

}

}

return prev;

}

list\* sssort(list\* prev)

{

BOOK \*sort, knijka;

int i,check;

list \*spisak;

sort=(BOOK \*)malloc(broi\*sizeof(BOOK));

for(i=0,spisak=prev;i<broi;spisak=spisak->next,i++)

{

sort[i].nali4nost=spisak->kniga.nali4nost;

sort[i].year=spisak->kniga.year;

sort[i].day=spisak->kniga.day;

sort[i].mount=spisak->kniga.mount;

strcpy(sort[i].number,spisak->kniga.number);

strcpy(sort[i].name,spisak->kniga.name);

strcpy(sort[i].izdatel,spisak->kniga.izdatel);

strcpy(sort[i].avtorime,spisak->kniga.avtorime);

strcpy(sort[i].avtorprezime,spisak->kniga.avtorprezime);

strcpy(sort[i].avtorfamiliq,spisak->kniga.avtorfamiliq);

}

printf("Za sortirane po data izberete 1, za sortirane po avtor izberete 2\n");

do{

scanf("%d",&i);

if(i==1||i==2) break;

else printf("\n\n\nVuvedete 1 ili 2 !!!\n\n");

}while(1);

switch(i)

{

case 2:

do{

check=1;

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

{

if(i<broi-1)

{

if(strcmp(sort[i].avtorime,sort[i+1].avtorime)>0)

{

knijka.nali4nost=sort[i].nali4nost;

knijka.year=sort[i].year;

knijka.day=sort[i].day;

knijka.mount=sort[i].mount;

strcpy(knijka.number,sort[i].number);

strcpy(knijka.name,sort[i].name);

strcpy(knijka.izdatel,sort[i].izdatel);

strcpy(knijka.avtorime,sort[i].avtorime);

strcpy(knijka.avtorprezime,sort[i].avtorprezime);

strcpy(knijka.avtorfamiliq,sort[i].avtorfamiliq);

sort[i].nali4nost=sort[i+1].nali4nost;

sort[i].year=sort[i+1].year;

sort[i].day=sort[i+1].day;

sort[i].mount=sort[i+1].mount;

strcpy(sort[i].number,sort[i+1].number);

strcpy(sort[i].name,sort[i+1].name);

strcpy(sort[i].izdatel,sort[i+1].izdatel);

strcpy(sort[i].avtorime,sort[i+1].avtorime);

strcpy(sort[i].avtorprezime,sort[i+1].avtorprezime);

strcpy(sort[i].avtorfamiliq,sort[i+1].avtorfamiliq);

sort[i+1].nali4nost=knijka.nali4nost;

sort[i+1].year=knijka.year;

sort[i+1].day=knijka.day;

sort[i+1].mount=knijka.mount;

strcpy(sort[i+1].number,knijka.number);

strcpy(sort[i+1].name,knijka.name);

strcpy(sort[i+1].izdatel,knijka.izdatel);

strcpy(sort[i+1].avtorime,knijka.avtorime);

strcpy(sort[i+1].avtorprezime,knijka.avtorprezime);

strcpy(sort[i+1].avtorfamiliq,knijka.avtorfamiliq);

check=0;

}

}

}

}while(!check);

do{

check=1;

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

{

if(i<broi-1)

{

if( (strcmp(sort[i].avtorprezime,sort[i+1].avtorprezime)>0) && (strcmp(sort[i].avtorime,sort[i+1].avtorime)<=0) )

{

knijka.nali4nost=sort[i].nali4nost;

knijka.year=sort[i].year;

knijka.day=sort[i].day;

knijka.mount=sort[i].mount;

strcpy(knijka.number,sort[i].number);

strcpy(knijka.name,sort[i].name);

strcpy(knijka.izdatel,sort[i].izdatel);

strcpy(knijka.avtorime,sort[i].avtorime);

strcpy(knijka.avtorprezime,sort[i].avtorprezime);

strcpy(knijka.avtorfamiliq,sort[i].avtorfamiliq);

sort[i].nali4nost=sort[i+1].nali4nost;

sort[i].year=sort[i+1].year;

sort[i].day=sort[i+1].day;

sort[i].mount=sort[i+1].mount;

strcpy(sort[i].number,sort[i+1].number);

strcpy(sort[i].name,sort[i+1].name);

strcpy(sort[i].izdatel,sort[i+1].izdatel);

strcpy(sort[i].avtorime,sort[i+1].avtorime);

strcpy(sort[i].avtorprezime,sort[i+1].avtorprezime);

strcpy(sort[i].avtorfamiliq,sort[i+1].avtorfamiliq);

sort[i+1].nali4nost=knijka.nali4nost;

sort[i+1].year=knijka.year;

sort[i+1].day=knijka.day;

sort[i+1].mount=knijka.mount;

strcpy(sort[i+1].number,knijka.number);

strcpy(sort[i+1].name,knijka.name);

strcpy(sort[i+1].izdatel,knijka.izdatel);

strcpy(sort[i+1].avtorime,knijka.avtorime);

strcpy(sort[i+1].avtorprezime,knijka.avtorprezime);

strcpy(sort[i+1].avtorfamiliq,knijka.avtorfamiliq);

check=0;

}

}

}

}while(!check);

do{

check=1;

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

{

if(i<broi-1)

{

if( (strcmp(sort[i].avtorfamiliq,sort[i+1].avtorfamiliq)>0) && (strcmp(sort[i].avtorprezime,sort[i+1].avtorprezime)<=0) && (strcmp(sort[i].avtorime,sort[i+1].avtorime)<=0) )

{

knijka.nali4nost=sort[i].nali4nost;

knijka.year=sort[i].year;

knijka.day=sort[i].day;

knijka.mount=sort[i].mount;

strcpy(knijka.number,sort[i].number);

strcpy(knijka.name,sort[i].name);

strcpy(knijka.izdatel,sort[i].izdatel);

strcpy(knijka.avtorime,sort[i].avtorime);

strcpy(knijka.avtorprezime,sort[i].avtorprezime);

strcpy(knijka.avtorfamiliq,sort[i].avtorfamiliq);

sort[i].nali4nost=sort[i+1].nali4nost;

sort[i].year=sort[i+1].year;

sort[i].day=sort[i+1].day;

sort[i].mount=sort[i+1].mount;

strcpy(sort[i].number,sort[i+1].number);

strcpy(sort[i].name,sort[i+1].name);

strcpy(sort[i].izdatel,sort[i+1].izdatel);

strcpy(sort[i].avtorime,sort[i+1].avtorime);

strcpy(sort[i].avtorprezime,sort[i+1].avtorprezime);

strcpy(sort[i].avtorfamiliq,sort[i+1].avtorfamiliq);

sort[i+1].nali4nost=knijka.nali4nost;

sort[i+1].year=knijka.year;

sort[i+1].day=knijka.day;

sort[i+1].mount=knijka.mount;

strcpy(sort[i+1].number,knijka.number);

strcpy(sort[i+1].name,knijka.name);

strcpy(sort[i+1].izdatel,knijka.izdatel);

strcpy(sort[i+1].avtorime,knijka.avtorime);

strcpy(sort[i+1].avtorprezime,knijka.avtorprezime);

strcpy(sort[i+1].avtorfamiliq,knijka.avtorfamiliq);

check=0;

}

}

}

}while(!check);

break;

case 1:

do{

check=1;

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

{

if(i<broi-1)

{

if(sort[i].year>sort[i+1].year)

{

knijka.nali4nost=sort[i].nali4nost;

knijka.year=sort[i].year;

knijka.day=sort[i].day;

knijka.mount=sort[i].mount;

strcpy(knijka.number,sort[i].number);

strcpy(knijka.name,sort[i].name);

strcpy(knijka.izdatel,sort[i].izdatel);

strcpy(knijka.avtorime,sort[i].avtorime);

strcpy(knijka.avtorprezime,sort[i].avtorprezime);

strcpy(knijka.avtorfamiliq,sort[i].avtorfamiliq);

sort[i].nali4nost=sort[i+1].nali4nost;

sort[i].year=sort[i+1].year;

sort[i].day=sort[i+1].day;

sort[i].mount=sort[i+1].mount;

strcpy(sort[i].number,sort[i+1].number);

strcpy(sort[i].name,sort[i+1].name);

strcpy(sort[i].izdatel,sort[i+1].izdatel);

strcpy(sort[i].avtorime,sort[i+1].avtorime);

strcpy(sort[i].avtorprezime,sort[i+1].avtorprezime);

strcpy(sort[i].avtorfamiliq,sort[i+1].avtorfamiliq);

sort[i+1].nali4nost=knijka.nali4nost;

sort[i+1].year=knijka.year;

sort[i+1].day=knijka.day;

sort[i+1].mount=knijka.mount;

strcpy(sort[i+1].number,knijka.number);

strcpy(sort[i+1].name,knijka.name);

strcpy(sort[i+1].izdatel,knijka.izdatel);

strcpy(sort[i+1].avtorime,knijka.avtorime);

strcpy(sort[i+1].avtorprezime,knijka.avtorprezime);

strcpy(sort[i+1].avtorfamiliq,knijka.avtorfamiliq);

check=0;

}

}

}

}while(!check);

do{

check=1;

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

{

if(i<broi-1)

{

if(sort[i].year==sort[i+1].year&&sort[i].mount>sort[i+1].mount)

{

knijka.nali4nost=sort[i].nali4nost;

knijka.year=sort[i].year;

knijka.day=sort[i].day;

knijka.mount=sort[i].mount;

strcpy(knijka.number,sort[i].number);

strcpy(knijka.name,sort[i].name);

strcpy(knijka.izdatel,sort[i].izdatel);

strcpy(knijka.avtorime,sort[i].avtorime);

strcpy(knijka.avtorprezime,sort[i].avtorprezime);

strcpy(knijka.avtorfamiliq,sort[i].avtorfamiliq);

sort[i].nali4nost=sort[i+1].nali4nost;

sort[i].year=sort[i+1].year;

sort[i].day=sort[i+1].day;

sort[i].mount=sort[i+1].mount;

strcpy(sort[i].number,sort[i+1].number);

strcpy(sort[i].name,sort[i+1].name);

strcpy(sort[i].izdatel,sort[i+1].izdatel);

strcpy(sort[i].avtorime,sort[i+1].avtorime);

strcpy(sort[i].avtorprezime,sort[i+1].avtorprezime);

strcpy(sort[i].avtorfamiliq,sort[i+1].avtorfamiliq);

sort[i+1].nali4nost=knijka.nali4nost;

sort[i+1].year=knijka.year;

sort[i+1].day=knijka.day;

sort[i+1].mount=knijka.mount;

strcpy(sort[i+1].number,knijka.number);

strcpy(sort[i+1].name,knijka.name);

strcpy(sort[i+1].izdatel,knijka.izdatel);

strcpy(sort[i+1].avtorime,knijka.avtorime);

strcpy(sort[i+1].avtorprezime,knijka.avtorprezime);

strcpy(sort[i+1].avtorfamiliq,knijka.avtorfamiliq);

check=0;

}

}

}

}while(!check);

do{

check=1;

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

{

if(i<broi-1)

{

if(sort[i].year==sort[i+1].year&&sort[i].mount==sort[i+1].mount&&sort[i].day>sort[i+1].day)

{

knijka.nali4nost=sort[i].nali4nost;

knijka.year=sort[i].year;

knijka.day=sort[i].day;

knijka.mount=sort[i].mount;

strcpy(knijka.number,sort[i].number);

strcpy(knijka.name,sort[i].name);

strcpy(knijka.izdatel,sort[i].izdatel);

strcpy(knijka.avtorime,sort[i].avtorime);

strcpy(knijka.avtorprezime,sort[i].avtorprezime);

strcpy(knijka.avtorfamiliq,sort[i].avtorfamiliq);

sort[i].nali4nost=sort[i+1].nali4nost;

sort[i].year=sort[i+1].year;

sort[i].day=sort[i+1].day;

sort[i].mount=sort[i+1].mount;

strcpy(sort[i].number,sort[i+1].number);

strcpy(sort[i].name,sort[i+1].name);

strcpy(sort[i].izdatel,sort[i+1].izdatel);

strcpy(sort[i].avtorime,sort[i+1].avtorime);

strcpy(sort[i].avtorprezime,sort[i+1].avtorprezime);

strcpy(sort[i].avtorfamiliq,sort[i+1].avtorfamiliq);

sort[i+1].nali4nost=knijka.nali4nost;

sort[i+1].year=knijka.year;

sort[i+1].day=knijka.day;

sort[i+1].mount=knijka.mount;

strcpy(sort[i+1].number,knijka.number);

strcpy(sort[i+1].name,knijka.name);

strcpy(sort[i+1].izdatel,knijka.izdatel);

strcpy(sort[i+1].avtorime,knijka.avtorime);

strcpy(sort[i+1].avtorprezime,knijka.avtorprezime);

strcpy(sort[i+1].avtorfamiliq,knijka.avtorfamiliq);

check=0;

}

}

}

}while(!check);

break;

}

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

{

printf("Библиотечен номер: %s",sort[i].number);

printf("\nИме: %s",sort[i].name);

printf("Издателство: %s",sort[i].izdatel);

if(sort[i].nali4nost=='y')

{

printf("Наличност: Взета");

}else if(sort[i].nali4nost=='n')

{

printf("Наличност: Налична");

}else if(sort[i].nali4nost=='b')

{

printf("Наличност: Бракувана");

}else if(sort[i].nali4nost=='z')

{

printf("Наличност: Запазена");

}

printf("\nДата на издаване: %d.%d.%d",sort[i].day,sort[i].mount,sort[i].year);

printf("\nАвтор: %s %s %s\n",sort[i].avtorime,sort[i].avtorprezime,sort[i].avtorfamiliq);

}

return prev;

}

int main(void)

{

int ch;

list\* prev=NULL;

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

for(;;)

{

printf ("1. Въвеждане на нова книга.\n");

printf ("2. Изтриване на книга.\n");

printf ("3. Изписване на информацията за книгата на екрана.\n");

printf ("4. Запис на данните за книгите в файл.\n");

printf ("5. Прочитане на данните за книгите от файл.\n");

printf ("6. Изписване на данните за всички книги на екрана.\n");

printf ("7. Информация за налични книги по даден автор.\n");

printf ("8. Сортиране по автор и дата на издаване.\n");

printf ("9. ИЗХОД \n");

printf ("Изберете опция от следното меню: ");

fflush(stdin);

ch=getche();

switch(ch)

{

case '1':

system("cls");

prev=addbook(prev);

system("Pause");

system("cls");

break;

case '2':

system("cls");

prev=dellbook(prev);

system("Pause");

system("cls");

break;

case '3':

system("cls");

prev=printone(prev);

system("Pause");

system("cls");

break;

case '4':

system("cls");

prev=zapis(prev);

system("Pause");

system("cls");

break;

case '5':

system("cls");

prev=readfile(prev);

system("Pause");

system("cls");

break;

case '6':

system("cls");

prev=printall(prev);

system("Pause");

system("cls");

break;

case '7':

system("cls");

prev=lookfor(prev);

system("Pause");

system("cls");

break;

case '8':

system("cls");

prev=sssort(prev);

system("Pause");

system("cls");

break;

case '9':

printf("\n\n");

system("Pause");

return 0;

default:

system("cls");

break;

}

}

}