



КУРСОВА ЗАДАЧА

ПО ПРОГРАМИРАНЕ

ТЕМА: СИСТЕМА ЗА УПРАВЛЕНИЕ И ОБРАБОТВАНЕ НА ДАННИ

ТЕМАТИЧНА ОБЛАСТ: БИБЛИОТЕКА

ЗАДАНИЕ

Да се проектира и реализира система за управление и обработване на данни No...(по избор като е допустима промяна на полетата).

Програмната система да осигурява въвеждане на нови данни, преглед на данните, обновяване на данните, допълване на данните, изтриване на записи, отпечатване на данни, сортиране по определено поле, както и справки по единични полета и комбинации от две полета.

Тематична област: Библиотека

Кат.No	Заглавие	Автор	Издателство	Година	Тип
--------	----------	-------	-------------	--------	-----



Структура на записите в основния файл

Файлът от определен тип е крайна наредена съвкупност от еднотипни данни на външен носител. Отделните елементи могат да бъдат от всякакъв тип с изключение от тип файл. За да се работи в една програма с файл трябва да се декларира файлова променлива.

1. Деклариране на променлива за файл от определен тип става по следния начин: име: file of тип; където име е името на файловата променлива, а тип е типа на елементите на файла.

DataFile:File of Book;

2. Писането във файл става с библиотечната процедура:
write(файлова_променлива, израз1, израз2, ..., изразN), където стойностите на израз1, израз2, ..., изразN са от типа на елементите на файла.

Write(DataFile,BaseData[i])

3. Четенето от файл става с библиотечната процедура
read(файлова_променлива, променлива1, променлива2, ...,променливаN), където променлива1, променлива2, ..., променливаN са от типа на елементите на файла.

Read(DataFile,BaseData[i]);

4. Затварянето на файл става с библиотечната процедура **procedure** close(f: **file of** тип), където f е файловата променлива.

Close(DataFile);

5. Осъществяване на връзка между името на файла на външния носител и файловата променлива става чрез библиотечна процедура: **procedure** assign(f: **file of** тип; s: **string**)

Assign(DataFile,'Biblioteka.dat');

6. Отварянето на нов файл става чрез библиотечната процедура: **procedure** rewrite(f: **file of** тип). Тази процедура създава нов, празен файл с име и място указани преди това в процедурата assign. Ако вече има файл с това име, той се унищожавя и се създава новия файл. В новия файл може само да се пише.

Rewrite(DataFile);

7. Отварянето на вече съществуващ файл става чрез библиотечната процедура: **procedure** reset(f: **file of** тип). Тази процедура отваря файл с име и място указани преди това в процедурата assign. В така отворения файл може и да се чете, и да се пише.

Reset (DataFile);

8. Когато се чете от един файл, при достигане края на файла трябва да се спре четенето. Проверката за край на файла става чрез библиотечната функция: **function** eof(f: **file of** тип):**boolean**; където f е файловата променлива.

```
While not eof(DataFile) do
  begin
    Inc(i);
    Read(DataFile,BaseData[i]);
  end;
```

Записът е крайна наредена структура от данни, състояща се от разнотипни елементи, наречени *полета*. Той служи да групира в себе си разнородна информация, описваща някакъв сложен информационен обект.



ТЕХНИЧЕСКИ УНИВЕРСИТЕТ – СОФИЯ СЕКТОР “СЛЕДДИПЛОМНО ОБУЧЕНИЕ”

В програмата е използван запис Book, състоящ се от следните полета:

```
KatNo :String[3];  
  Zagl:String[25];  
  Avtor:String[15];  
  Izdatel:String[10];  
  Tip:String[10];  
  Godizd:String[4];  
  Cena:real;
```

Използвана е библиотека CRT, заради функциите:

- **Clrscr** - Изчиства екрана.
- **Gotoxy(int,int)** - Позиционира курсора по зададени координати.
- **Textbackground(word/int)** - Цвят на фона.
- **Textcolor(word/int)** - Цвят на текста.
- **Readkey** - Чете натиснат клавиш от клавиатурата.

В програмата са създадени следните процедури, които да бъдат извиквани за изпълнение в зависимост от нуждите на потребителя на програмната система:

- | | |
|---------------------------------|------------------------------------|
| 1. WriteRK | 9. ZaglQuery |
| 2. ReadData | 10. CenaQuery |
| 3. WriteData | 11. ZaglAvtorQuery |
| 4. GetRecord | 12. Queries |
| 5. AddRecords | 13. LookAt |
| 6. ShowRecord | 14. SortByCena |
| 7. DeleteRecord | 15. SortByKatNo |
| 8. IzdatelQuery | 16. PrintRecords |

Функционалните възможности на програмата

Главното меню на информационната система е организирано чрез оператора за многовариантен избор Case, като всеки избор на число/ от 1 до 9/ води до изпълнението на предварително програмирани процедури.

- Избирането на числото „1” стартира процедурата LookAt , която позволява да се прелистят въведените данни/ напред, назад, първи и последен запис/, да се направят корекции или изтриване на запис, и връщане в основното меню.
- Избирането на числото „2” стартира процедурата Queries. Тя е организирана като подменю, позволяващо изпълнението на процедурите за търсене и извеждане на записи по определени критерии, в случая: по автор, по заглавие, по цена и по заглавие и автор. Позволява връщане в основно меню.



ТЕХНИЧЕСКИ УНИВЕРСИТЕТ – СОФИЯ СЕКТОР “СЛЕДДИПЛОМНО ОБУЧЕНИЕ”

- Избирането на числото „3” стартира процедурата AddRecord, която позволява въвеждането на нови записи, като след всеки се дава възможност за въвеждане на следващ или край на въвеждането.
- Избирането на числото „4” стартира процедурата за изтриване на запис, като първо се извежда съобщение за номера на записа за изтриване и след това се изпълнява процедурата DeleteRecord.
- Избирането на числото „5” стартира процедурата за коригиране на данните на запис. Показва записа за редактиране и позволява да се въведат нови данни за книгата.
- Избирането на числото „6” стартира процедурата SortByCena, която позволява подреждането по цена на книгите във възходящ ред.
- Избирането на числото „7” стартира процедурата SortByKatNo, която сортира по каталожен номер записите.
- Избирането на числото „8” стартира процедурата PrintRecords, позволяваща отпечатване на данните.
- Избирането на числото „9” стартира процедурата изход от програмата

Листинг на програмата и резултати

```
Program Biblioteka;  
Uses Crt,Printer,Dos;  
Type Book=record  
    KatNo      :String[3];
```



```
Zagl      :String[25];
Avtor     :String[15];
Izdatel   :String[10];
Tip       :String[10];
Godizd    :String[4];
Cena      :real;
end;
Const MaxNumOfBook=800;
Var
  DataFile:File of Book;
  BaseData:array [1..MaxNumOfBook] of Book;
  I,Last :Integer;
  Ch,Choice:Char;
  year,month,day,DoW,H, Min, Sec, Sec100 :word;
{-----}
Procedure WriteRK(i,j:byte;St:string);
begin
  GotoXY(j,i); write(St);
end;
{-----}
Procedure ReadData;
begin { ReadData }
  {$I-}
  Reset (DataFile);
  {$I+}
  if IOResult<>0 then
    begin
      Rewrite(DataFile);
      Close(DataFile);
      Reset(DataFile);
    end; {endif}
  i:=0;
  While not eof(DataFile) do
    begin
      Inc(i);
      Read(DataFile,BaseData[i]);
    end; {endwhile}
  Last:=i;
end; { ReadData }
{-----}
Procedure WriteData;
begin { WriteData }
  Rewrite(DataFile);
  for i := 1 to Last do
    write(DataFile,BaseData[i]);
  end; { WriteData }
{-----}
```



Procedure GetRecord(Var R:Book);

```
begin { GetRecord }  
  writeRK(5,10,'KatNo      :'); ReadLn(R.KatNo);  
  writeRK(7,10,'Zaglavie   :'); ReadLn(R.Zag);  
  writeRK(9,10,'Avtor      :'); ReadLn(R.Avtor);  
  writeRK(11,10,'Izdatel   :'); ReadLn(R.Izdatel);  
  writeRK(13,10,'Tip        :'); ReadLn(R.Tip);  
  writeRK(15,10,'Godina     :'); ReadLn(R.Godizd);  
  writeRK(17,10,'Cena       :'); ReadLn(R.Cena);  
end; { GetRecord }
```

{-----}

Procedure AddRecords;

```
var  
  Ch:char;  
begin { AddRecords }  
  Repeat  
    Inc(last);  
    ClrScr;  
    WriteRK(3,15,'Vavedete danni za kniga No '); Write(Last);  
    GetRecord(BaseData[Last]);  
    WriteRK(19,15,'Shte vavejdate li oshte zapisi? <Y/N>');  
    repeat  
      Ch:=readkey;  
    until Ch in ['y','Y','n','N'];  
  Until Ch in ['n','N'];  
end; { AddRecords }
```

{-----}

Procedure ShowRecord(i:integer);

```
begin { ShowRecord }  
  ClrScr;  
  writeRK(3,10,'Poreden No  :'); writeln(i);  
  writeRK(5,10,'KatNo       :'); Write(BaseData[i].KatNo);  
  writeRK(7,10,'Zaglavie    :'); Write(BaseData[i].Zagl);  
  writeRK(9,10,'Avtor       :'); Write(BaseData[i].Avtor);  
  writeRK(11,10,'Izdatel    :'); Write(BaseData[i].Izdatel);  
  writeRK(13,10,'Tip         :'); Write(BaseData[i].Tip);  
  writeRK(15,10,'Godina      :'); Write(BaseData[i].Godizd);  
  writeRK(17,10,'Cena        :'); Write(BaseData[i].Cena:2:2);  
end; { ShowRecord }
```

{-----}

Procedure DeleteRecord(var k:integer);

```
begin { DeleteRecord }  
  ShowRecord(k);  
  Writeln;  
  Writeln('Zalichavane na danni za zapis s poreden nomer ',k);  
  Writeln;  
  Write('    Potvardete <Y/N>');
```



```
repeat
  Ch:=readkey;
until Ch in ['y','Y','n','N'];
if Ch In ['y','Y']
then
begin
  While (k<Last) do
  begin
    BaseData[k]:=BaseData[k+1];
    Inc(k);
  end;
  Dec(Last);
  Writeln ('Dannite sa zalicheni');
end {then}
else Writeln('Dannite ne sa zalicheni');
Ch:=ReadKey;
end; { DeleteRecord }
{-----}
Procedure IzdatelQuery;
var F:String[10];
    Ch:Char;
begin
  ClrScr;
  WriteRK(5,15,'Vavedete izdatelstvo:   ');
  Readln(F);
  i:=1;
  Repeat
    if F=BaseData[i].Izdatel then
    begin
      ShowRecord(i);
      WriteRK(22,5,' Natisnete proizvolen klavish - sledvasht zapis;   Esc - otkaz ');
      Ch:=ReadKey;
    end;
    Inc(i);
  Until (i>Last) or (ord(Ch)=27);
end;
{-----}
Procedure ZaglQuery;
var F: String[25];
    Ch:Char;
begin { StateQuery }
  ClrScr;
  WriteRK(5,15,'Vavedete zaglavie na kniga:   ');
  Readln(F);
  i:=1;
  Repeat
    if F=BaseData[i].Zagl then
```



```
begin
  ShowRecord(i);
  WriteRK(22,5,' Natisnete proizvolen klavish - sledvasht zapis;   Esc - otkaz ');
  Ch:=ReadKey;
end;
Inc(i);
Until (i>Last) or (ord(Ch)=27);
end; { StateQuery }
{-----}
Procedure CenaQuery;
var Min,Max : LongInt;
    Ch :Char;
begin { SalaryQuery }
  ClrScr;
  WriteRK(3,15,'Vavedete dolna i gorna granica na cenata');
  WriteRK(5,15,'Minimum :   '); Readln(Min);
  WriteRK(7,15,'Maximum :   '); Readln(Max);
  i:=1;
  Repeat
    if (BaseData[i].Cena>=Min) and (BaseData[i].Cena<=Max) then
      begin
        ShowRecord(i);
        WriteRK(22,5,'Natisnete proizvolen klavish - sledvasht zapis;   Esc - otkaz ');
        Ch:=ReadKey;
      end;
    Inc(i);
  Until(i>Last) or (ord(Ch)=27);
end; { SalaryQuery }
{-----}
Procedure ZaglAvtorQuery;
var F : string[25];
    P : string[15];
    Ch: Char;
begin
  ClrScr;
  WriteRK(3,10,'Vavedete zaglavieto na knigata:   '); Readln(F);
  WriteRK(5,10,'Vavedete avtora:   '); Readln(P);
  i:=1;
  Repeat;
  if (BaseData[i].Zagl=F) and (BaseData[i].Avtor=P) then
    begin
      ShowRecord(i);
      WriteRK(22,5,' Natisnete proizvolen klavish - sledvasht zapis;   Esc - otkaz ');
      Ch:=ReadKey;
    end;
  Inc(i);
  Until(i>Last) or (ord(Ch)=27);
```



```
end;  
{-----}
```

Procedure Queries;

```
var  
  Ch:Char;  
begin { Queries }  
  Repeat  
    ClrScr;  
    TextBackGround(10);  
    for i:=3 to 21 do  
      writeRK(i,15,'                ');  
    TextColor(14);  
    WriteRK(3,30,' Tyrsene');  
    WriteRK(6,15,' Po izdatel..... 1 ');  
    WriteRK(8,15,' Po zaglavie.....2 ');  
    WriteRK(10,15,' Po cena.....3 ');  
    WriteRK(12,15,' Po zaglavie i avtor.....4 ');  
    WriteRK(14,15,' Glavno menu.....5 ');  
    TextBackGround(18);  
    TextColor(15);  
    writeRK(19,20,' Изберете < 1...5 > ');  
    repeat  
      Ch:=ReadKey;  
    until Ch in ['1','2','3','4','5'];  
    Case Ch of  
      '1':IzdatelQuery;  
      '2':ZaglQuery;  
      '3':CenaQuery;  
      '4':ZaglAvtorQuery;  
    end; {Case}  
    until Ch='5';  
  end; { Queries }  
{-----}
```

Procedure LookAt;

```
begin { LookAt }  
  i:=1;  
  Repeat  
    ShowRecord(i);  
    TextColor(14);  
    writeRK(19,20,'Enter                - Korekciq na zapisa);  
    writeRK(20,20,'Esc                - Izhod');  
    writeRK(21,20,'Del                - Zalichavane na danni');  
    writeRK(22,20,'Left(Down or PageDown) - Sledvastha kniga');  
    writeRK(23,20,'Righ(Up or PageUp)   - Predhodna kniga');  
    writeRK(24,20,'Home                - Pytvi zapis');  
    writeRK(25,20,'End                - Posleden zapis');  
  repeat
```



```
Ch:=ReadKey;
until ord(Ch) in [083,079,071,072,080,077,075,073,081,13,27];
if (ord(Ch) in[080,075,081]) and (i<Last) then Inc(i);
if (ord(Ch) in[072,077,073]) and (i>1) then Dec(i);
if ord(Ch)=083 then DeleteRecord(i);
if ord(Ch)=079 then i:=Last;
if ord(Ch)=071 then i:=1;
if ord(Ch)=13 then GetRecord(BaseData[i]);
Until ord(Ch)=27;
end; { LookAt }
{-----}
Procedure SortByCena;
var Flag:boolean;
    P :Book;
begin { SortByCod }
ClrScr;
WriteRK(1,15,Sortirane na dannii po pole cena – vyzhod.red');
repeat
    Flag:=True;
    for i := 1 to Last-1 do
        if BaseData[i].Cena>BaseData[i+1].Cena then
            begin
                Flag:=False;
                P:=BaseData[i]; BaseData[i]:=BaseData[i+1]; BaseData[i+1]:=P;
            end;
    until Flag;
WriteRK(3,10,'Dannite sa sortirani po pole cena');
WriteRK(5,15,'Natisni klavish za glavno menu');
Ch:=ReadKey;
end; { SortByCod }
{-----}
Procedure SortByKatNo;
var Flag:boolean;
    P :Book;
begin { SortByCod }
ClrScr;
WriteRK(1,15,Sortirane na dannii po pole katalogen nomer');
repeat
    Flag:=True;
    for i := 1 to Last-1 do
        if BaseData[i].KatNo>BaseData[i+1].KatNo then
            begin
                Flag:=False;
                P:=BaseData[i]; BaseData[i]:=BaseData[i+1]; BaseData[i+1]:=P;
            end;
    until Flag;
WriteRK(3,10,'Dannite sa sortirani po pole katalogen nomer');
```



ТЕХНИЧЕСКИ УНИВЕРСИТЕТ – СОФИЯ

СЕКТОР “СЛЕДДИПЛОМНО ОБУЧЕНИЕ”

```
WriteRK(5,15,'Natisni klavish za glavno menu');
Ch:=ReadKey;
end; { SortByCod }
{-----}
Procedure PrintRecords;
var N,K: integer;
begin { PrintRecords }
  ClrScr;
  writeRK(3,15,'Отпечатване на запис No: '); readln(N);
  writeRK(5,27,'до запис No: '); readln(K);
  for i := N to K do
  begin
    {$I-}
    write(Lst,i:8);
    {$I+}
    if IOResult <>0 then Exit;
    write(Lst,BaseData[i].KatNo:3);
    write(Lst,BaseData[i].Zagl:25);
    write(Lst,BaseData[i].Avtor:15);
    write(Lst,BaseData[i].Izdatel:10);
    write(Lst,BaseData[i].Tip:10);
    write(Lst,BaseData[i].Godizd:4);
    write(Lst,BaseData[i].Cena:2:2);
    writeln(Lst);
  end;
end; { PrintRecords }
{-----}
      { Г Л А В Н А   П Р О Г Р А М А }
```

```
begin
  ClrScr;
  Assign(DataFile,'Biblioteka.dat');
  Last:=0;
  ReadData;
  repeat
    ClrScr;
    GetDate(Year, Month, Day, DoW);
    write('Dnes e: ',Day,'/',Month,'/',Year,'g. ');
    Case DoW of
      1: Write(' ponedelnik');
      2: Write(' vtornik');
      3: Write(' srqda');
      4: Write(' chetvartak');
      5: Write(' petak');
      6: Write(' sabota');
      7: Write(' nedelq');
    end;
  end;
```



ТЕХНИЧЕСКИ УНИВЕРСИТЕТ – СОФИЯ

СЕКТОР “СЛЕДДИПЛОМНО ОБУЧЕНИЕ”

```
GetTime(H, Min, Sec, Sec100);
GotoXY(66,1);write(H,':',Min,':',Sec);write('h. ');
TextBackGround(18);
for i:=3 to 20 do
  writeRK(i,15,' ');
TextColor(14);
writeRK(3,25,'BIBLIOTEKA “IVAN VAZOV”');
writeRK(5,25,' KATALOG');
writeRK(8,15,'Knigi-pregled..... 1');
writeRK(10,15,'Spravki.....2');
writeRK(12,15,'Dobavqne na nova kniga.....3');
writeRK(14,15,'Iztrivane na danni za kniga.....4');
writeRK(16,15,'Korekciq na dannite. ....5');
writeRK(18,15,'Sortirane po cena.....6');
writeRK(20,15,'Sortirane po katalogen nomer.....7');
writeRK(22,15,'Otpechatvane na danni.....8');
writeRK(24,15,'Krai na rabotata.....9');

TextBackGround(18);
TextColor(15);
writeRK(26,25,' Изберете < 1...9 > ');
repeat
  Choice:=ReadKey;
until Choice in['1','2','3','4','5','6','7','8','9'];

Case Choice of
  '1':LookAt;
  '2':Queries;
  '3':AddRecords;
  '4': begin ClrScr;
        writeRK(3,10,'Nomer na zapis za otstranqvanе? ');
        readln(i);
        DeleteRecord(i);
        end;
  '5':begin ClrScr;
        writeRK(3,10,'Nomer na zapis za redaktirane? ');
        readln(i);
        ShowRecord(i);
        writeln;
        writeln(' Novi danni za knjigata');
        GetRecord(BaseData[i]);
        end;
  '6':SortByCena;
  '7':SortByKatNo;
  '8':PrintRecords;
end; {endcase}
Until Choice='9';
```



```
WriteData;  
Close(DataFile);  
ClrScr;  
end.
```