**Лабораторная работа №2  
Типизированный файл, записи, перечисляемый тип**

**Поздняков Юрий А-04-19, Вар. 14**

**1. Условие задачи**

В типизированном файле, хранящем записи вида **С1**, найти (и вывести) **С2**, и переставить её по правилу **С3**.

В столбце ***Тип***: п – перечисляемый, с – строковый, ч – числовой, \* – любой

Дата – *TDataTime*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Тип** | **С1** | **С2** | **С3** |
| 14 | п  с  с | Существо (вампир, оборотень и т.д.)  Имя  Место рождения | *Последнюю* запись о существе с не менее чем двумя буквами «А» в имени | С *первой* записью файла |

**2. Уточненное условие задачи**

В типизированном файле, хранящем записи вида: Creation (Vampire, Werevolf, Elf, Ork) (Перечисляемый тип), FirstName (Строка из 30 символов), BornPlace (Строка из 30 символов) найти и вывести последнюю запись о существе с не менее чем двумя буквами «A» (англ.) в FirstName и переставить её с первой записью файла.

Для *создания* типизированного файла предусмотреть две возможности:   
а) считывание данных из заранее подготовленного нестандартного текстового файла, б) добавление данных в типизированный файл в режиме диалога.

Предусмотреть также возможность *просмотра* данных из типизированного файла.

Для написания своей программы воспользуйтесь *примером программы*, приведенным ниже. Для выбора нужного действия в программе использовано *меню*.

**3. Пример**

Пусть заданы данные по шаблону (<Creation> <FirstName> <BornPlace>):

**0 Eklipsis Mordor**

**1 Kirk Enterprise**

**2 Legalas GreatWood**

**3 Molag Ball**

**2 USS Kelvin**

В ней запись номер 3 содержит две буквы «A» (англ.). Переставляем ее с записью номер 1:

**2 Legalas GreatWood**

**1 Kirk Enterprise**

**0 Eklipsis Mordor**

**3 Molag Ball**

**2 USS Kelvin**

**3. Комментарии**

Комментарии расставлены в программе. Язык интерфейса – английский (во избежание ошибок кодировок и упрощения международного понимания)

**4. Тесты**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Исходные данные | Ожидаемый результат | Смысл теста |
| 1 | ‘’ | ‘Not found!’ | Аномалия 1 Без входных данных |
| 2 | ‘-1 namee placee’ | ‘Not found!’ | Аномалия 2 Индекс вышел за границы, пустая строка |
| 3 | ‘1 Ficus Libba’ | ‘Not found!’ | Нужных элементов не найдено |
| 4 | ‘2 Jabba Krio’ | 'Needed Creation is first!’ | Найден нужный элемент, но он является первым и он единственный |
| 5 | ‘2 Jabba Krio  1 Ficus Libba’ | 'Needed Creation is first!’ | Найден нужный элемент, но он является первым |
| 6 | ‘1 Ficus Libba  2 Jabba Krio’ | ‘Founded! Jabba (Elf, krio)  Replaced (2 and 1)’ | Найден нужный элемент, поменялся местами с первым |
| 7 | ‘1 Ficus Libba  2 Jabba Krio  3 kirAAA night’ | ‘Founded! kirAAA (Ork, night’)  Replaced (3 and 1)’ | Найден нужный элемент в конце и последний (еще есть Jabba), поменялся местами с первым с игнорированием регистра |
| 8 | ‘0 Eklipsis Mordor  1 Kirk Enterprise  2 Legalas GreatWood  3 Molag Ball  2 USS Kelvin’ | ‘Founded! Legalas (Elf, GreatWood’)  Replaced (3 and 1)’ | Найден нужный элемент, поменялся местами с первым - тест из примера |
| 9 | ‘1 Ficus Libba  2 kirAAA night  3 Jabba Krio‘ | ‘Founded! kirAAA (Ork, night’)  Replaced (2 and 1)’ | Найден нужный элемент из середины, поменялся местами с первым с игнорированием регистра |

**5. Код**

**Программа:**

Program Lab2; // 1) T-FIle, 2) Text File

{$APPTYPE CONSOLE}

Uses SysUtils, //Libs

Windows,

UnitLab2;

Var ch: char;

Begin

SetConsoleCP(1251);

SetConsoleOutputCP(1251); /Russian Lang???

Writeln('-----===== LAB #2 =====-----');

//Menu system

Repeat

Writeln('--------------------------------------');

Writeln('1 - Create T-File from dialog');

Writeln('2 - Create T-File from text-file');

Writeln('3 - Search and edit');

Writeln('4 - View');

Writeln('5 - Exit');

Write('Your choice: ');

Readln(ch);

Writeln('--------------------------------------');

Case ch of

'1': CreateTypedFile1; //create dialog

'2': CreateTypedFile2; //create from file

'3': FindAndEdit; //search

'4': ViewFile; //view

'5': Exit; //exit

else

Begin

Writeln('Wrong command!');

Write('Press ENTER...');

Readln;

end;

end;

Until ch = '5';

end.

**Модуль:**

Unit UnitLab2;

Interface

Uses SysUtils; //libs

Type

TCreation = (Vampire, Werevolf, Elf, Ork); // 0 - 3

TCharacter = Record

Creation : TCreation;

FirstName : String[30];

BornPlace : String[30];

end;

//----------create dialog

Procedure CreateTypedFile1();

//----------create from file

Procedure CreateTypedFile2();

//--------------search

Procedure FindAndEdit();

//--------------get count of char in word

Function GetCharCount(st : String; ch : Char) : Integer;

//--------------get text of TCreation

Function GetTitleOfCreation(tc : TCreation) : String;

//--------------view

Procedure ViewFile();

Implementation

//----------create dialog

Procedure CreateTypedFile1();

Var

Character: TCharacter;

f : File of TCharacter;

c, k : Integer;

ch : Char;

FileName : string[80];

Begin

if ParamCount < 1 then

Begin

Writeln('Params < 1'#13#10'Press ENTER...');

Readln;

Exit;

end;

FileName := ParamStr(1);

AssignFile(f, FileName);

Try Rewrite(f);

Except

Writeln('Error: Cannot create T-File: ', FileName);

Write('Press ENTER...');

Readln;

Exit;

end;

c := 0;

Repeat //Input from keyboard

Write('Creation (Vampire-0, Werevolf-1, Elf-2, Ork-3): '); Readln(k); Character.Creation := TCreation(k);

Write('FirstName: '); Readln(Character.FirstName);

Write('BornPlace: '); Readln(Character.BornPlace);

Write(f, Character);

Inc(c);

Write('More? (y/n): '); //More

Readln(ch);

Until LowerCase(ch) = 'n';

CloseFile(f);

Writeln('T-File created, count: ', c);

Write('Press ENTER...');

Readln;

end;

//----------create from file

Procedure CreateTypedFile2();

Var

Character : TCharacter;

ft : TextFile;

f : File of TCharacter;

i1, c : Integer;

s1, s2, s3 : String;

FileName1, FileName2 : string[80];

Begin

if ParamCount < 2 then

Begin

Writeln('Params < 2'#13#10'Press ENTER...');

Readln;

Exit;

end;

FileName1 := ParamStr(1); //out

FileName2 := ParamStr(2); //in

AssignFile(ft, FileName2);

Try Reset(ft);

Except

Writeln('Error: Cannot open T-File: ', FileName2);

Write('Press ENTER...');

Readln;

Exit;

end;

AssignFile(f, FileName1);

Try Rewrite(f);

Except

Writeln('Error: Cannot create T-File: ', FileName1);

Write('Press ENTER...');

Readln;

Exit;

end;

c := 0;

i1 := 0;

While not EOF(ft) do

Begin

//<INTEGER>

//<STRING>

//<STRING>

Readln(ft, s1);

Readln(ft, s2);

Readln(ft, s3);

Try i1 := StrToInt(s1); //read all fields

Except

Writeln('Error: Cannot parse to integer: ', s1);

Write('Press ENTER...');

Readln;

Exit;

end;

Character.Creation := TCreation(i1); //creating element

Character.FirstName := s2;

Character.BornPlace := s3;

Write(f, Character);

//Writeln('#', c, ': ', Character.FirstName, ' (', GetTitleOfCreation(Character.Creation), ', ', Character.BornPlace, ')'); //FOR TESTS

Inc(c);

end;

if c > 0 then Writeln('T-File created, count: ', c) //Count

else Writeln('No data in file!');

CloseFile(ft);

CloseFile(f);

Write('Press ENTER...');

Readln;

end;

//--------------search

Procedure FindAndEdit();

Var

Character, Found : TCharacter;

f : File of TCharacter;

c, fc : Integer;

FileName : string[80];

Begin

if ParamCount < 1 then

Begin

Writeln('Params < 1'#13#10'Press ENTER...');

Readln;

Exit;

end;

FileName := ParamStr(1);

AssignFile(f, FileName);

Try Reset(f);

Except

Writeln('Error: Cannot open T-File: ', FileName);

Write('Press ENTER...');

Readln;

Exit;

end;

fc := -1;

c := 1;

While not EOF(f) do

Begin

Read(f, Character);

if (GetCharCount(Character.FirstName, 'A') >= 2) then //two or more AA in word

Begin

Found := Character; //FOUND!

fc := c;

end;

Inc(c);

end;

if fc <> -1 then

Begin

if fc = 1 then Writeln ('Needed Creation is first!') else

Begin //Output result

Writeln('Founded! ', Found.FirstName, ' (', GetTitleOfCreation(Found.Creation), ', ', Found.BornPlace, ') ');

Seek(f, 0);

Read(f, Character);

Seek(f, 0); //Replacing

Write(f, Found);

Seek(f, fc - 1);

Write(f, Character);

Writeln('Replaced (', fc, ' and 1)');

end;

end

else

Begin

Writeln('Not found!');

end;

CloseFile(f);

Write('Press ENTER...');

Readln;

end;

Function GetCharCount(st : String; ch : Char) : Integer;

Var c, i : Integer;

Begin

c := 0;

ch := UpCase(ch);

for i := 1 to Length(st) do

if UpCase(st[i]) = ch then Inc(c);

GetCharCount := c;

end;

Function GetTitleOfCreation(tc : TCreation) : String;

Var st : String;

Begin

Case tc of

Vampire: st := 'Vampire';

Werevolf: st := 'Werevolf';

Elf: st := 'Elf';

Ork: st := 'Ork';

end;

GetTitleOfCreation := st;

end;

//--------------view

Procedure ViewFile();

Var

Character : TCharacter;

f : File of TCharacter;

n, c : Integer;

FileName : string[80];

Begin

if ParamCount < 1 then

Begin

Writeln('Params < 1'#13#10'Press ENTER...');

Readln;

Exit;

end;

FileName := ParamStr(1);

AssignFile(f, FileName);

Try Reset(f);

Except

Writeln('Error: Cannot open T-File: ', FileName);

Write('Press ENTER...');

Readln;

Exit;

end;

c := 0;

While not EOF(f) do

Begin

Read(f, Character);

Writeln('#', c, ': ', Character.FirstName, ' (', GetTitleOfCreation(Character.Creation), ', ', Character.BornPlace, ')');

Inc(c);

end;

if c = 0 then

Writeln('No data in file!')

else

Writeln('Count: ', c);

CloseFile(f);

Write('Press ENTER...');

Readln;

end;

End.