**РУСЕНСКИ УНИВЕРСИТЕТ „АНГЕЛ КЪНЧЕВ“**

**КУРСОВА РАБОТА**

ПО

СОФТУЕРНО ИНЖЕНЕРСТВО

Студент:

Фак. Номер:

Група:

Специалност:

Дата: Проверил:

# Съдържание

[1. Задание 3](#_Toc344907667)

[2. Диаграми 3](#_Toc344907668)

[2.1 Delphi диаграма 3](#_Toc344907671)

[2.2 Use Case 4](#_Toc344907672)

[2.3 Sequence 5](#_Toc344907673)

[2.4 Activity 6](#_Toc344907674)

[2.5 Class 7](#_Toc344907675)

[3. Работа с програмата 8](#_Toc344907676)

[4. Особености на програмата 12](#_Toc344907677)

[5. Код на програмата 13](#_Toc344907678)

[5.1 Unit1.pas 13](#_Toc344907679)

[5.2 Unit2.pas 15](#_Toc344907680)

[5.3 Unit3.pas 16](#_Toc344907681)

# Задание

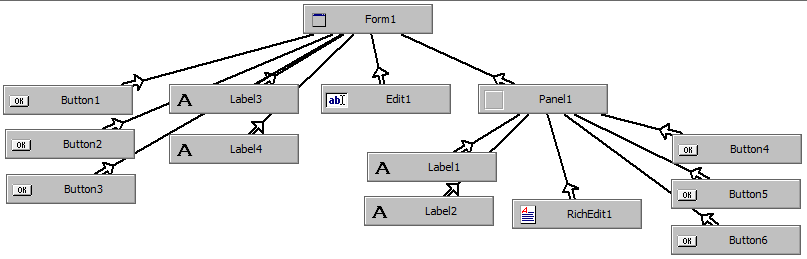
Да се проектира и реализира Windows приложение за обслужване на индексно-последователен файл. Инструменталното средство да представлява терминологичен речник с минимум 25 команди на Delphi и да предоставя на потребителя следните функционални възможности:

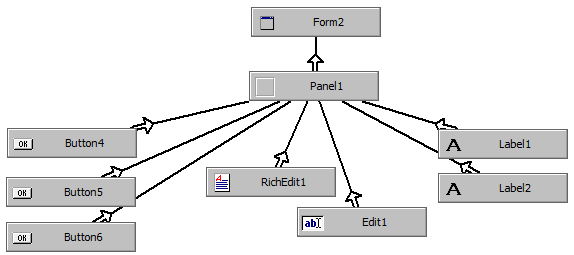
* Добавяне, редактиране и изтриване на команда;
* Търсене на команда;
* Логически контрол на всички полета;
* Генериране на отчет с подадените команди, дата и време за начало и край на сеанса.

# Диаграми

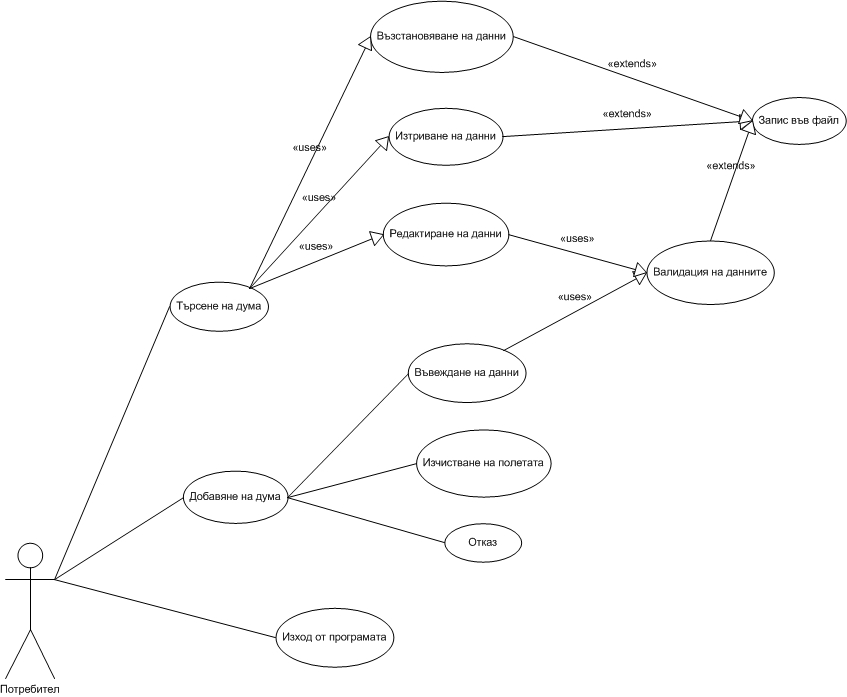


## Delphi диаграма

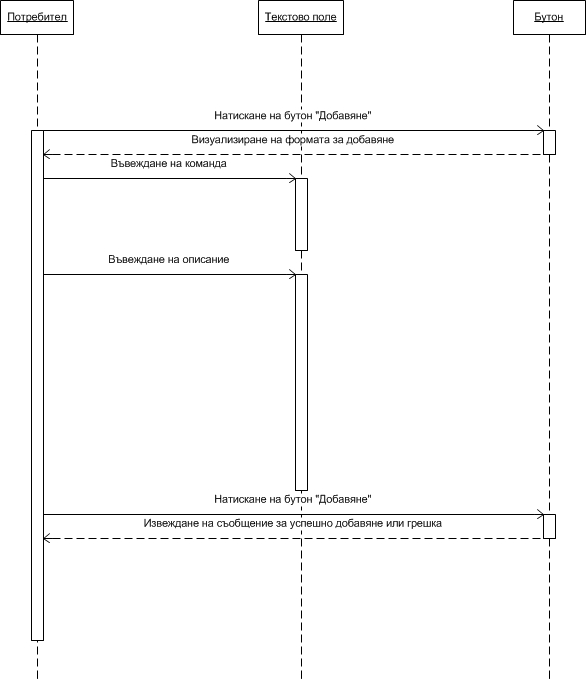




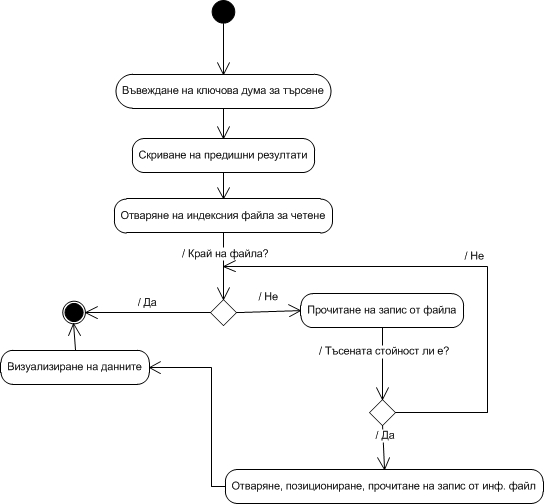
## Use Case



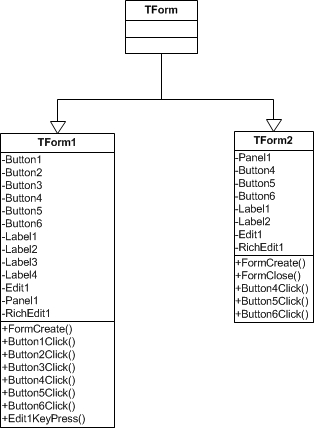
## Sequence



## Activity

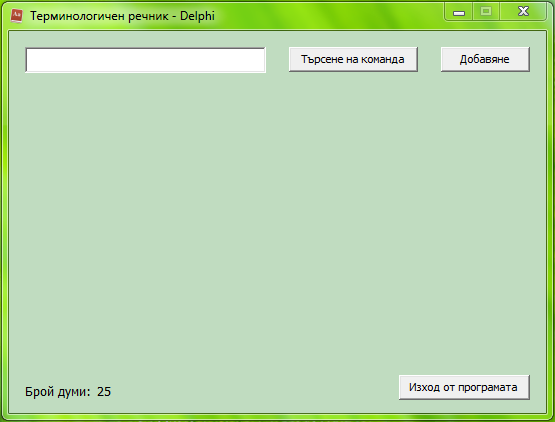


## Class



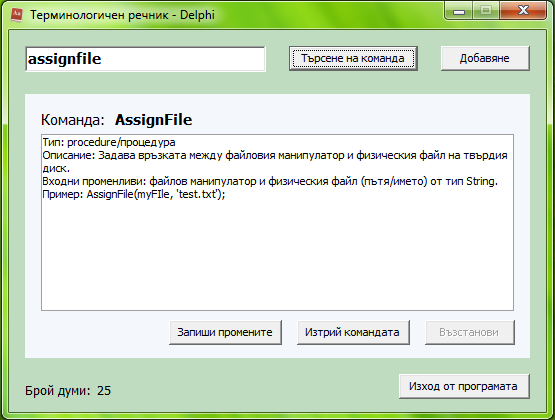
# Работа с програмата

След стартиране на програмния файл \*.ехе на екрана се показва прозорът от фиг. 1.



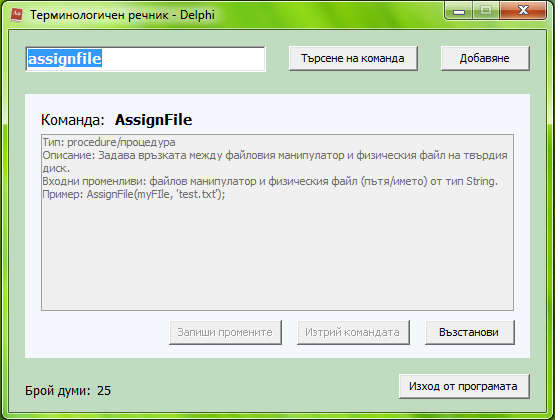
Фиг. 1 Терминологичен речник – Delphi

В текстовото поле се въвежда думата за търсене, след което се натиска или бутонът „Търсене на команда“, или Enter. Ако думата я няма не става нищо, в противен случай се появява описанието за нея (фиг. 2). Важно е индексният и информационният файл да са предварително създадени в същата директория, в която е програмата и съответно да бъдат със следните имена: „index.ind“ и „commands.txt“. Ако не са създадени, то програмата информира за това, чрез съобщение.



Фиг. 2 Търсене на ключ

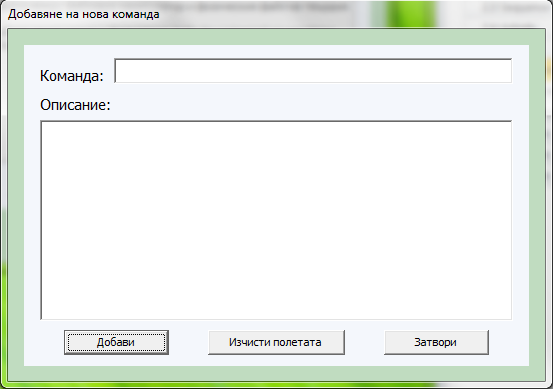
Виждат се три нови бутона. Първият „Запиши промените“ служи за записване на данните след като са редактирани. Вторият „Изтрий командата“ служи за логическо изтриване на дума (команда), т.е данните не се изтриват физически от файла/вете (фиг. 3).



Фиг. 3 Логически изтрит запис (дума/команда)

Когато думата е „изтрита“ (фиг. 3) не е възможно редактирането на данните, но тя може да бъде възстановена чрез бутона „Възстанови“ и след това да се редактира.

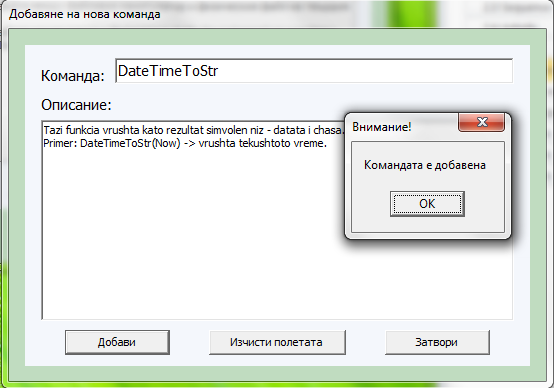
В долния ляв ъгъл на прозореца се показва броят на записаните думи в речника. В случая са 25. Добавянето на нова дума става като се кликне върху бутон „Добавяне“. Появява се нов прозорец (фиг. 4).



Фиг. 4 Форма за добавяне

В първото поле се въвежда командата, а във второто – описание, примери и т.н.

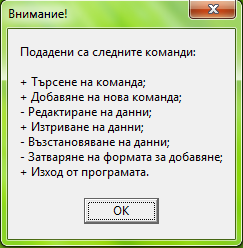
След въвеждането, за да се добави новата дума във файла се кликва върху „Добави“. Ако полетата не са празни се появява съобщението от фиг. 5.



Фиг. 5 Добавяне на нова дума

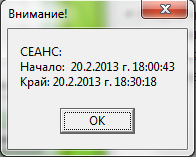
След което прозорецът автоматично се затваря. Вторият бутон „Изчисти полетата“ от фиг. 4 служи за изчистване на полетата, а „Затвори“ – за затваряне на прозореца за добавяне без да се записват данните от полетата във файла, ако има такива.

За изход от програмата се изпозлва бутонът „Изход от програмата“ (фиг. 1). След като се избере този бутон се появяват следните прозорци от фиг. 6 и фиг 7. В първия са изборени всички операции, които може да изпълнява програмата. Пред тези операции има или -, или +. (-) – показва, че операцията не е избрана, а (+) – обратното (фиг. 6).



фиг. 6 Използвани операции

Във втория прозорец се показва информацията за датата и времето за начало и край на сеанса(фиг. 7).



фиг. 7 Начало и край на сеанса

# Особености на програмата

* Размер – около 800 KB;
* ОС – Windows XP, 7, 8;
* Програмата е тествана с ОС Windows 7 Ultimate x64 и Windows 8 Professional x64 при резолюция на екрана 1920x1080 px.

# Код на програмата

## Unit1.pas

unit Unit1;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,

Dialogs, StdCtrls, ComCtrls, ExtCtrls;

type

TForm1 = class(TForm)

Edit1: TEdit;

Button1: TButton;

Button2: TButton;

Panel1: TPanel;

Button3: TButton;

Label1: TLabel;

Label2: TLabel;

Button4: TButton;

Button5: TButton;

RichEdit1: TRichEdit;

Button6: TButton;

Label3: TLabel;

Label4: TLabel;

procedure FormCreate(Sender: TObject);

procedure Button3Click(Sender: TObject);

procedure Button2Click(Sender: TObject);

procedure Button1Click(Sender: TObject);

procedure Button4Click(Sender: TObject);

procedure Edit1KeyPress(Sender: TObject; var Key: Char);

procedure Button6Click(Sender: TObject);

procedure Button5Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form1: TForm1;

beginSeans, EndSeans: string;

implementation

uses Unit2, Unit3;

{$R \*.dfm}

procedure TForm1.FormCreate(Sender: TObject);

begin

searchCom := '-';

addCom := '-';

closeF1 := '-';

closeF2 := '-';

editCom := '-';

deleteCom := '-';

restoreCom := '-';

beginSeans := DateTimeToStr(Now);

Panel1.Visible := false;

AssignFile(indF, 'index.ind');

AssignFile(infF, 'commands.txt');

{$I-} Reset(infF); {$I+}

if ioresult <> 0 then

ShowMessage('Грешка при отваряне на информационния файл!')

else begin

{$I-} Reset(indF); {$I+}

if ioresult <> 0 then

ShowMessage('Грешка при отваряне на индексния файл!')

else

Label4.Caption := IntToStr(FileSize(infF));

end;

CloseFile(infF); CloseFile(indF);

end;

procedure TForm1.Button3Click(Sender: TObject);

begin

closeF1 := '+';

ShowMessage('Подадени са следните команди:' + #13#10 +

#13#10 + searchCom + ' Търсене на команда;' +

#13#10 + addCom + ' Добавяне на нова команда;' +

#13#10 + editCom + ' Редактиране на данни;' +

#13#10 + deleteCom + ' Изтриване на данни;' +

#13#10 + restoreCom + ' Възстановяване на данни;' +

#13#10 + closeF2 + ' Затваряне на формата за добавяне;' +

#13#10 + closeF1 + ' Изход от програмата.');

endSeans := DateTimeToStr(Now);

ShowMessage('СЕАНС:' + #13#10 + 'Начало: ' + beginSeans + #13#10 + 'Край: ' + endSeans);

Form1.Close;

end;

procedure TForm1.Button2Click(Sender: TObject);

begin

Form2.Show;

end;

procedure TForm1.Button1Click(Sender: TObject);

begin

searchCom := '+';

if Edit1.Text='' then begin

Panel1.Visible := false;

ShowMessage('Не сте въвели команда/ключ за търсене!');

end

else begin

Panel1.Visible := false;

positionFound := Search(LowerCase(Edit1.Text));

if (positionFound <> -1) then begin

Panel1.Visible := true;

{$I-} Reset(infF); {$I+}

if ioresult <> 0 then

ShowMessage('Грешка при отваряне на информационния файл!')

else begin

Seek(infF, positionFound);

Read(infF, komanda);

CloseFile(infF);

Label2.Caption := komanda.komanda;

RichEdit1.Lines.Text := komanda.opisanie;

if komanda.deleted = true then begin

Button4.Enabled := false;

Button5.Enabled := false;

Button6.Enabled := true;

RichEdit1.Enabled := false;

end

else begin

Button4.Enabled := true;

Button5.Enabled := true;

Button6.Enabled := false;

RichEdit1.Enabled := true;

end;

end;

end;

end;

end;

procedure TForm1.Button4Click(Sender: TObject);

begin

editCom := '+';

if (RichEdit1.Text = '') then

ShowMessage('Въведете описание!')

else begin

{$I-} Reset(infF); {$I+}

if ioresult <> 0 then

ShowMessage('Грешка при отваряне на информационния файл!')

else begin

Seek(infF, positionFound);

komanda.opisanie := RichEdit1.Lines.Text;

write(infF, komanda);

ShowMessage('Данните за "' + komanda.komanda + '" са променени!');

CloseFile(infF);

end;

end;

end;

procedure TForm1.Edit1KeyPress(Sender: TObject; var Key: Char);

begin

if Key = #13 then

Button1.OnClick(Sender);

end;

procedure TForm1.Button6Click(Sender: TObject);

begin

restoreCom := '(+)';

{$I-} Reset(infF); {$I+}

if ioresult <> 0 then

ShowMessage('Грешка при отваряне на информационния файл!')

else begin

Reset(infF);

Seek(infF, positionFound);

komanda.deleted := false;

write(infF, komanda);

ShowMessage('Командата "' + komanda.komanda + '" е възстановена!');

CloseFile(infF);

Button4.Enabled := true;

Button5.Enabled := true;

Button6.Enabled := false;

RichEdit1.Enabled := true;

end;

end;

procedure TForm1.Button5Click(Sender: TObject);

begin

deleteCom := '+';

{$I-} Reset(infF); {$I+}

if ioresult <> 0 then

ShowMessage('Грешка при отваряне на информационния файл!')

else begin

Reset(infF);

Seek(infF, positionFound);

komanda.deleted := true;

write(infF, komanda);

ShowMessage('Командата "' + komanda.komanda + '" е изтрита!');

CloseFile(infF);

Button4.Enabled := false;

Button5.Enabled := false;

Button6.Enabled := true;

RichEdit1.Enabled := false;

end;

end;

end.

## Unit2.pas

unit Unit2;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,

Dialogs, StdCtrls, ComCtrls, ExtCtrls;

type

TForm2 = class(TForm)

Panel1: TPanel;

Label1: TLabel;

Button4: TButton;

Button5: TButton;

Button6: TButton;

Edit1: TEdit;

RichEdit1: TRichEdit;

Label2: TLabel;

procedure Button4Click(Sender: TObject);

procedure Button5Click(Sender: TObject);

procedure Button6Click(Sender: TObject);

procedure FormClose(Sender: TObject; var Action: TCloseAction);

procedure FormCreate(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form2: TForm2;

implementation

uses Unit1, Unit3;

{$R \*.dfm}

procedure TForm2.Button4Click(Sender: TObject);

begin

addCom := '+';

if (Edit1.Text = '') then

ShowMessage('Моля въведете команда!')

else if(RichEdit1.Lines.Capacity = 0) then

ShowMessage('Моля въведете описание!')

else begin

komanda.komanda := Edit1.Text;

komanda.opisanie := RichEdit1.Lines.Text;

komanda.deleted := false;

{$I-} Reset(infF); {$I+}

if ioresult <> 0 then

ShowMessage('Грешка при отварянето на информационния файл!')

else begin

{$I-} Reset(indF); {$I+}

if ioresult <> 0 then

ShowMessage('Грешка при отварянето на индексния файл!')

else begin

Seek(infF, FileSize(infF));

Write(infF, komanda);

Index.index := LowerCase(Edit1.Text);

Index.position := FileSize(infF);

Form1.Label4.Caption := IntToStr(FileSize(infF));

CloseFile(infF);

Seek(indF, FileSize(indF));

write(indF, Index);

CloseFile(indF);

SortIndexFile();

ShowMessage('Командата е добавена');

Form2.Close;

end;

end;

end;

end;

procedure TForm2.Button5Click(Sender: TObject);

begin

Edit1.Text := '';

RichEdit1.Lines.Clear;

end;

procedure TForm2.Button6Click(Sender: TObject);

begin

closeF2 := '+';

Form2.Close;

end;

procedure TForm2.FormClose(Sender: TObject; var Action: TCloseAction);

begin

Button5.OnClick(Sender);

end;

procedure TForm2.FormCreate(Sender: TObject);

begin

Button5.OnClick(Sender);

end;

end.

## Unit3.pas

unit Unit3;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,

Dialogs, StdCtrls, ComCtrls;

type

TIndex = Record

index : string[255];

position : integer;

end;

IndFile = File of TIndex;

TKomanda = Record

komanda: string[255];

opisanie: string[255];

deleted: boolean;

end;

InfFile = File of TKomanda;

var

indF: IndFile;

infF: InfFile;

komanda: TKomanda;

index: TIndex;

positionFound: integer;

{podadeni komandi}

searchCom, addCom, closeF1, closeF2,

editCom, deleteCom, restoreCom: string;

{end podadeni komandi}

procedure SortIndexFile();

function Search(keyword: string): integer;

implementation

procedure SortIndexFile();

var

M,J,H : Integer;

Buffer : TIndex;

IndexArray : array[1..100] of TIndex;

begin

{$I-} Reset(indF); {$I+}

if ioresult <> 0 then

ShowMessage('Грешка при отварянето на индексния файл!')

else begin

for J := 1 to FileSize(IndF) do

Read(IndF,IndexArray[J]);

H := FileSize(IndF) Div 2;

while H > 0 do begin

M := H + 1;

repeat

J := M - H;

while J > 0 do begin

if IndexArray[J].index <= IndexArray[J+H].index then

J := 0

else begin

Buffer := IndexArray[J];

IndexArray[J] := IndexArray[J + H];

IndexArray[J + H] := Buffer;

end;

J := J - H;

end;

M := M + 1;

until M > FileSize(IndF);

H := H div 2;

end;

Seek(IndF, 0);

for J := 1 to FileSize(IndF) do

Write(IndF,IndexArray[J]);

CloseFile(IndF);

end;

end;

function Search(keyword: string): integer;

var

I,L,D:Integer;

begin

{$I-} Reset(indF); {$I+}

if ioresult <> 0 then begin

ShowMessage('Грешка при отварянето на индексния файл!');

Result := -1;

exit;

end

else begin

if FileSize(indF) = 0 then begin

Result := -1;

exit;

end;

{cycles := 0;}

L := 0; D := FileSize(indF)-1;

repeat

{cycles := cycles + 1; }

I := (L + D) div 2;

Seek(indF, I);

Read(indF, Index);

if keyword > Index.index then

L := I + 1

else if keyword < Index.index then

D := I - 1;

until (keyword = Index.index) or (L > D);

if (keyword = Index.index) then

Result := Index.position-1

else

Result := -1;

end;

end;

end.