Міністерство освіти і науки України

Центральноукраїнський національний технічний університет

Механіко-технологічний факультет

ЗВІТ

ПРО ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ № 1

з навчальної дисципліни “Модульне програмування”

РЕАЛІЗАЦІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

НА ОСНОВІ МОДУЛІВ FPC

ВИКОНАВ

студент академічної групи КІ-17

\_\_\_\_\_\_\_\_\_\_ С.О. Мельник

ПЕРЕВІРИВ

старший викладач кафедри кібербезпеки та програмного забезпечення

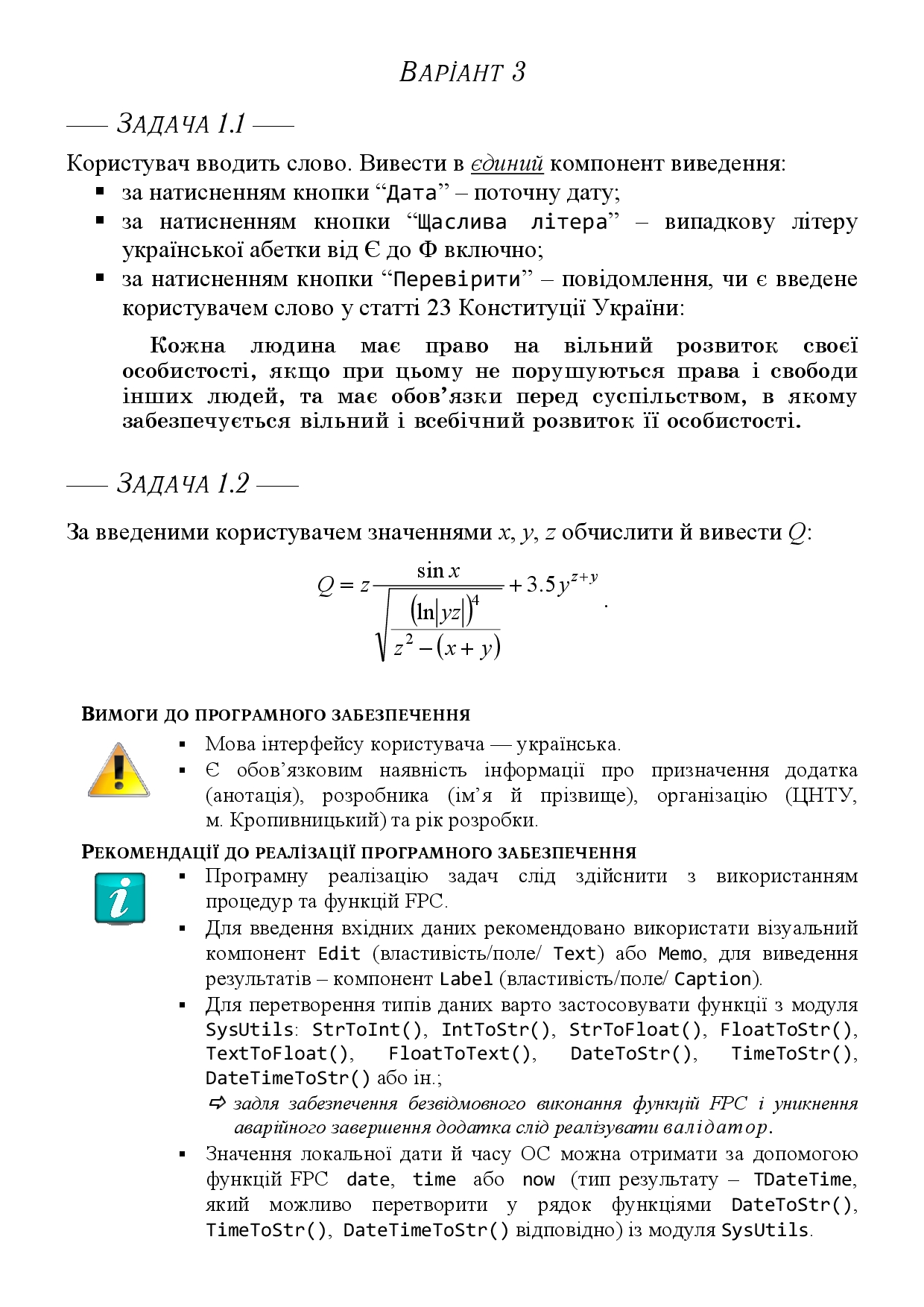
\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Л.І. Поліщук

Кропивницький – 2018

**Мета лабораторної роботи**

Полягає у набутті ґрунтовних вмінь і практичних навичок застосування методології модульного проектування програмного забезпечення, міжнародного стандарту ISO/IEC 12207, підходів до тестування програмних засобів на системному рівні, реалізації графічних додатків з застосуванням модулів FPC для оброблення текстових даних і математичних обчислень в кросплатформовому Lazarus IDE.

**Завдання до лабораторної роботи**

1. Реалізувати програмне забезпечення (графічний додаток) розв’язування задачі 1.1 на основі модулів FPC для оброблення символьних і рядкових даних.
2.  Реалізувати програмне забезпечення (графічний додаток) розв’язування задачі 1.2 на основі модулів FPC для математичних обчислень.

**Задача 1.1**

Аналіз задачі:

Дана задача має виконувати наступні функції:

- При натисканні кнопки «Дата» у полі виводу відображається дата.

- При натисканні кнопки «Щаслива літера» - випадкову літеру української абетки від Є до Ф

- При натисканні кнопки «Перевірити» - чи є введене слово у статті 23 Конституції України

Строга постановка задачі:

Вихідні дані: Edit.Text;

Вхідні дані: Edit.Text;

Обмеження та допущення: Edit.Text є типом String;

Аналіз вимог до програмного забезпечення:

-Інтерфейс користувача повинен відображатися виключно українською мовою.

-Інтерфейс має включати в себе наступну інформацію:

Додаток забезпечує:

-при натисканні кнопки "Дата" виводиться дата у форматі ДД.ММ.РРРР.

-- При натисканні кнопки «Щаслива літера» - випадкову літеру української абетки від Є до Ф

- При натисканні кнопки «Перевірити» - чи є введене слово у статті 23 Конституції України

ЦНТУ, м.Кропивницький, 2018 рік

Створив: Мельник Сергій Олександрович

Проектування архітектури програмного забезпечення:

Програма включає у себе робочий модуль, та форму.

Робочий модуль включає у себе процедуру, яка спрацьовує при натисканні кнопки «Дата», процедуру, яка спрацьовує при натисканні кнопки «Щаслива літера» та процедуру, яка спрацьовую при натисканні кнопки «Перевірити».

Процедура, яка спрацьовує при натисканні кнопки «Перевірка», що вміщає у себе підпрограму, яка перевіряє наявність рядка у вірші.

Ця підпрограма у свою чергу включає у себе код, який реагує на наявність рядка і код, який реагує на його відсутність.

Детальне проектування програмного забезпечення:

На формі повинні бути створено: 3 кнопки («Дата», «Щаслива літера» та «Перевірити»), 1 поле виведення

Процедура, що спрацьовує при натисканні кнопки «Дата» має забезпечувати вивід дати у форматі ДД.ММ.РРРР.

Процедура, що спрацьовує при натисканні кнопки «Щаслива літера» Має забезпечити вивід випадкової літери української абетки.

Процедура, що спрацьовує при натисканні кнопки «Перевірити» має забезпечувати перевірку наявності слові у статті, при цьому вона повинна нехтувати пунктуаційними чи граматичними помилками.

Код, що відповідає за реагування на наявність рядка у вірші повинен забезпечити вивід форми з відповідним повідомленням.

Код, що відповідає за реагування на відсутність рядка у вірші також повинен забезпечити вивід форми з відповідним повідомленням.

Вихідний код проекта lab1\_task1\_Melnyk

unit Unit1;

{$mode objfpc}{$H+}

interface

uses

Classes, SysUtils, FileUtil, Forms, Controls, Graphics, Dialogs, StdCtrls, Validator;

type

{ TForm1 }

TForm1 = class(TForm)

Button1: TButton;

Button2: TButton;

Button3: TButton;

Edit1: TEdit;

Label1: TLabel;

Label2: TLabel;

procedure Button1Click(Sender: TObject);

procedure Button2Click(Sender: TObject);

procedure Button3Click(Sender: TObject);

procedure Edit1Change(Sender: TObject);

procedure Label1Click(Sender: TObject);

procedure Label2Click(Sender: TObject);

private

public

end;

var

Form1: TForm1;

implementation

{$R \*.lfm}

{ TForm1 }

procedure TForm1.Label1Click(Sender: TObject);

begin

end;

procedure TForm1.Edit1Change(Sender: TObject);

begin

end;

procedure TForm1.Button1Click(Sender: TObject);

var

rand: integer;

bukvi: array [0..17] of string = ( 'Є', 'Ж', 'З', 'И', 'І', 'Ї', 'Й',

'К', 'Л', 'М', 'Н', 'О', 'П', 'Р', 'С', 'Т', 'У', 'Ф');

begin

randomize;

rand:=random(17);

Edit1.Text:=bukvi[rand];

end;

procedure TForm1.Button2Click(Sender: TObject);

var mon, day, year : word;

begin

DecodeDate (Date, year, mon, day);

Edit1.Text := IntToStr(day) + '.' + IntToStr(mon) + '.' + IntToStr(year);

end;

procedure TForm1.Button3Click(Sender: TObject);

var str, wd : string;

bool : boolean;

y : byte;

begin

str := Label1.Caption;

wd := Edit1.Text;

bool := Valid(wd);

if (bool = true) then begin

y := pos(wd, str);

if y > 0 then Edit1.Text := 'Слово присутнє!'

else Edit1.Text := 'Слово відсутнє!';

end

else Edit1.Text := 'Некоректно введені дані';

end;

procedure TForm1.Label2Click(Sender: TObject);

begin

end;

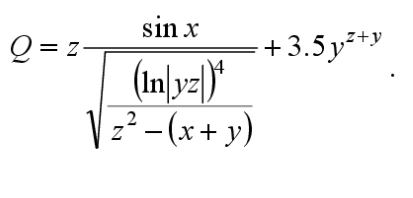
end.

**Задача 1.2**

Аналіз задачі:

Дана задача має виконувати наступні функції:

- При натисканні кнопки «Обчислити» виконується обчислення значення Q за формулою:



Строга постановка задачі:

Вихідні дані: Edit4.Text;

Вхідні дані: Edit1.Text; Edit2.Text; Edit3.Text;

Обмеження та допущення: уся введена і виведена інформація є типом String;

Аналіз вимог до програмного забезпечення:

-Інтерфейс користувача повинен відображатися виключно українською мовою.

-Інтерфейс має включати в себе наступну інформацію:

ЦНТУ

м.Кропивницький

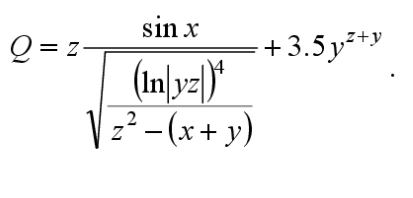
2018 рік

Створив: Мельник Сергій Олександрович

Додаток забезпечує:

-при натисканні кнопки " Обчислити" виконуються обчислення

по заданій формулі:



Проектування архітектури програмного забезпечення:

Програма включає у себе валідатор, робочий модуль, та форму.

Робочий модуль включає у себе процедуру, яка спрацьовує при натисканні кнопки «Обчислити».

Детальне проектування програмного забезпечення:

На формі повинні бути створено: 1 ( «Обчислити»), 8 полів виведення /введення (три під ввід даних, три під напис «Введіть х,y,z», одне для виведення результату та одне для виведення інформації про автора та функцій програмного забезпечення).

Процедура, яка спрацьовує при натисканні кнопки «Обчислити» має забезпечити вивід результату обчислення заданої функції.

Валідатор повинен відсікати недопустимі введені дані.

Вихідний код проекта lab1\_task2\_Melnyk

unit unit2;

{$mode objfpc}{$H+}

interface

uses

Validator1\_2, Math, Classes, SysUtils, FileUtil, Forms, Controls, Graphics, Dialogs, StdCtrls,

ExtCtrls;

type

{ TForm1 }

TForm1 = class(TForm)

Button1: TButton;

Edit1: TEdit;

Edit2: TEdit;

Edit3: TEdit;

Edit4: TEdit;

Image1: TImage;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Label4: TLabel;

Label5: TLabel;

Label6: TLabel;

procedure Button1Click(Sender: TObject);

procedure Edit1Change(Sender: TObject);

procedure Edit2Change(Sender: TObject);

procedure Edit3Change(Sender: TObject);

procedure Edit4Change(Sender: TObject);

procedure FormCreate(Sender: TObject);

procedure Image1Click(Sender: TObject);

procedure Label1Click(Sender: TObject);

procedure Label2Click(Sender: TObject);

procedure Label3Click(Sender: TObject);

procedure Label4Click(Sender: TObject);

procedure Label5Click(Sender: TObject);

procedure Label6Click(Sender: TObject);

private

public

end;

var

Form1: TForm1;

implementation

{$R \*.lfm}

{ TForm1 }

procedure TForm1.FormCreate(Sender: TObject);

begin

end;

procedure TForm1.Image1Click(Sender: TObject);

begin

end;

procedure TForm1.Button1Click(Sender: TObject);

var x, y, z : String;

q,t,r,x1, y1, z1: real;

i:byte;

//x1, y1, z1: integer;

begin

x:=Edit1.Text;

y:=Edit2.Text;

z:=Edit3.Text;

i:= valid1(x,y,z);

if (i=0) then

begin

Edit4.Text:='Некоректно введені дані.';

exit;

end;

x1:=strtofloat(x);

y1:= strtofloat(y);

z1:= strtofloat(z);

t:= (sin(x1)) / (sqrt( Power(ln(abs(y1\*z1)),4)));

r:= (sqr(z1)-(x1+y1));

i:= valid2(z1,x1,y1);

if (i=0) then

begin

Edit4.Text:='Некоректно введені дані.';

exit;

end;

q := z1\*(t/r)+3.5;

Edit4.Text := FloatToStr(q);

end;

procedure TForm1.Edit1Change(Sender: TObject);

begin

end;

procedure TForm1.Edit2Change(Sender: TObject);

begin

end;

procedure TForm1.Edit3Change(Sender: TObject);

begin

end;

procedure TForm1.Edit4Change(Sender: TObject);

begin

end;

procedure TForm1.Label1Click(Sender: TObject);

begin

end;

procedure TForm1.Label2Click(Sender: TObject);

begin

end;

procedure TForm1.Label3Click(Sender: TObject);

begin

end;

procedure TForm1.Label4Click(Sender: TObject);

begin

end;

procedure TForm1.Label5Click(Sender: TObject);

begin

end;

procedure TForm1.Label6Click(Sender: TObject);

begin

end;

end.

unit Validator1\_2;

{$mode objfpc}{$H+}

interface

function Valid1(var x,y,z: string):byte;

function Valid2(z,x,y: real):byte;

implementation

function Valid1(var x,y,z: string):byte;

var i,d,t:byte;

g,p:set of char ;

begin

g:=['0','1','2','3','4','5','6','7','8','9'];

p:=[',','-'];

valid1:=1;

d:=1;

t:=1;

if (x='')or(y='')or(z='') then

begin

valid1:=0 ;

exit;

end;

while (pos('.',x)<>0) do

x[pos('.',x)]:=',';

while (pos('.',y)<>0) do

y[pos('.',y)]:=',';

while (pos('.',z)<>0) do

z[pos('.',z)]:=',';

for i:=1 to length(x) do

begin

if not(x[i] in g)and not(x[i] in p) then

begin

valid1:=0;

exit;

end;

if ((x[i] in p)and (((x[i]=',')and (d=0)) or ((x[i]='-')and(t=0))) or ((i<>1) and (x[i]='-')))then

begin

valid1:=0;

exit;

end

else

begin

if (x[i]=',') then

begin

d:=0;

end;

if(x[i]='-') then

t:=0;

end;

end;

d:=1;

t:=1;

for i:=1 to length(y) do

begin

if not(y[i] in g)and not(y[i] in p) then

begin

valid1:=0;

exit;

end;

if ((y[i] in p)and (((y[i]=',')and (d=0)) or ((y[i]='-')and(t=0))) or ((i<>1) and (y[i]='-')))then

begin

valid1:=0;

exit;

end

else

begin

if (y[i]=',') then

begin

d:=0;

end;

if(y[i]='-') then

t:=0;

end;

end;

d:=1;

t:=1;

for i:=1 to length(z) do

begin

if not(z[i] in g)and not(z[i] in p) then

begin

valid1:=0;

exit;

end;

if ((((z[i]=',')and(d=0))or((z[i]='-')and(t=0))) or ((i<>1) and (z[i]='-')))then

begin

valid1:=0;

exit;

end

else

begin

if (z[i]=',') then

begin

d:=0;

end;

if(z[i]='-') then

t:=0;

end;

end;

end;

function Valid2(z,x,y: real):byte;

begin

valid2:=1;

if ((z\*z -(x+y))=0) then

valid2:=0 ;

end;

end.

**Висновок**

Перед тим, як виконувати лабораторну роботу №1 були оброблені теоретичні відомості, підготовані відповіді на контрольні запитання, та опрацьована самостійна підготовка до виконання лабораторної роботи №1, а також інформація подану на лекції.

Першою проблемою стала робота з репозиторієм GitHub, а саме з його налаштуванням. Витративши трохи часу все ж таки налаштування були виконаними.

Далі почалося виконання завдань з лабораторної роботи. По-перше була виконана задача №1. Модульний підхід до розробки ПЗ не викликав великих труднощів, навпаки він здався набагато зручнішим ніж процедурний. Найбільшою ж проблемою стало кодування символів форми, а саме кириличні символи. Щоб виправити усі помилки пов’язані з ним були витрачені 6 годин. Надалі розпочалося виконання задачі №2.

У ході виконання лабораторної роботи були здобуті навички створювати та використовувати модулі на мові Free Pascal, а також вивчений сам принцип модульної реалізації ПЗ. Застосування модулів виявилося дуже зручним, адже їх можна використовувати багатократно у різних програмних засобах. Дана лабораторна робота подарувала безцінний досвід роботи з модулями.

Загалом, завдання до цієї лабораторної роботи були досить цікавими, адже вони провели ознайомлення з модульним підходом реалізації ПЗ, зауваження відсутні.