Кузнецов Антон Денисович, ИВТ 1.1, КУРС 2

***Лабораторная работа 1***

## Тема: Введение. Знакомство с интегрированной средой Lazarus

**Постановка задачи:** Написать процедуру обработки события - щелчок по кнопке - сложение двух чисел. Для этого необходимо создать процедуру обработки события (двойным щелчком по объекту кнопка), объявить переменные, ввести значение переменных *a* и *b,*вычислить значение переменной  *с*и вывести значение *с* на экран.

**Код программы:**

unit Unit1;

{$mode objfpc}{$H+}

interface

uses

Classes, SysUtils, LResources, Forms, Controls, Graphics, Dialogs, StdCtrls,

Buttons;

type

{ TForm1 }

TForm1 = class(TForm)

Button1: TButton;

Edit1: TEdit;

Edit2: TEdit;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

procedure Button1Click(Sender: TObject);

private

{ private declarations }

public

{ public declarations }

end;

var

Form1: TForm1;

implementation

{ TForm1 }

procedure TForm1.Button1Click(Sender: TObject);

begin

Label1.Caption:= IntToStr(StrToInt(Edit1.Text)+StrToInt(Edit2.Text));

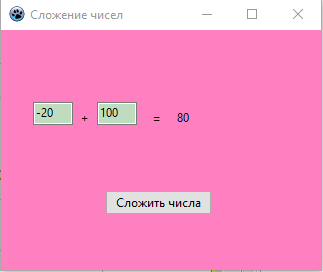
end;

initialization

{$I unit1.lrs}

end.

**Скриншот программы:**



***Лабораторная работа 2***

## Тема: Работа с графикой

**Постановка задачи:** Нарисовать в объекте**TImage** треугольник, одна сторона которого черная, другая красная и третья - синяя.

**Код программы:**

unit Unit1;

{$mode objfpc}{$H+}

interface

uses

Classes, SysUtils, FileUtil, Forms, Controls, Graphics, Dialogs, ExtCtrls,

StdCtrls;

type

{ TForm1 }

TForm1 = class(TForm)

Button1: TButton;

Edit1: TEdit;

Edit2: TEdit;

Edit3: TEdit;

Edit4: TEdit;

Edit5: TEdit;

Edit6: TEdit;

Image1: TImage;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Label4: TLabel;

Label5: TLabel;

Label6: TLabel;

procedure Button1Click(Sender: TObject);

procedure Image1Click(Sender: TObject);

private

{ private declarations }

public

{ public declarations }

end;

var

Form1: TForm1;

implementation

{$R \*.lfm}

{ TForm1 }

procedure TForm1.Image1Click(Sender: TObject);

begin

end;

procedure TForm1.Button1Click(Sender: TObject);

begin

Image1.Canvas.Brush.Color:= clwhite;

Image1.Canvas.pen.Color := clwhite;

Image1.Canvas.Rectangle (481, 313, 0, 0);

Image1.Canvas.LineTo(StrtoInt(Edit2.Text),StrtoInt(Edit1.Text));

Image1.Canvas.pen.Color := clblack;

Image1.Canvas.LineTo(StrtoInt(Edit4.Text),StrtoInt(Edit3.Text));

Image1.Canvas.pen.Color := clred;

Image1.Canvas.LineTo(StrtoInt(Edit6.Text),StrtoInt(Edit5.Text));

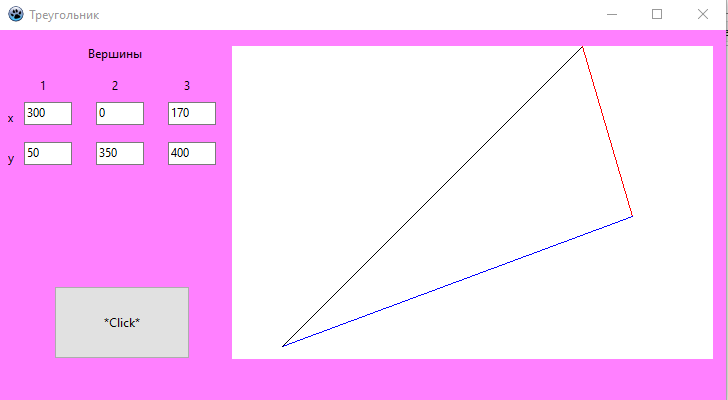
Image1.Canvas.pen.Color := clblue;

Image1.Canvas.LineTo(StrtoInt(Edit2.Text),StrtoInt(Edit1.Text));

end;

end.

**Скриншот программы:**



***Лабораторная работа 3***

## Тема: Построение графика заданной функции в указанном диапазоне

**Постановка задачи:** Построить график функции y = Ax^2 + Bx + C в диапазоне ( x min , y min ) – ( x max , y max ). Постоянные А, В, С, x min , y min , x max , y max вводятся через пользовательский интерфейс.

**Код программы:**

unit Unit1;

{$mode objfpc}{$H+}

interface

uses

Classes, SysUtils, FileUtil, Forms, Controls, Graphics, Dialogs, ExtCtrls,

StdCtrls;

type

{ TForm1 }

TForm1 = class(TForm)

Button1: TButton;

Edit1: TEdit;

Edit2: TEdit;

Edit3: TEdit;

Edit4: TEdit;

Edit5: TEdit;

Image1: TImage;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Label4: TLabel;

Label5: TLabel;

procedure Button1Click(Sender: TObject);

private

{ private declarations }

public

{ public declarations }

end;

var

Form1: TForm1;

i : integer;

x, y : real;

implementation

{$R \*.lfm}

{ TForm1 }

procedure TForm1.Button1Click(Sender: TObject);

begin

Image1.Canvas.Brush.Color:= clwhite;

Image1.Canvas.pen.Color := clwhite;

Image1.Canvas.Rectangle (424, 472, 0, 0);

Image1.Canvas.LineTo(212,0);

Image1.Canvas.pen.Color := clblack;

Image1.Canvas.LineTo(212,472);

Image1.Canvas.pen.Color := clwhite;

Image1.Canvas.LineTo(0,236);

Image1.Canvas.pen.Color := clblack;

Image1.Canvas.LineTo(424,236);

x := StrtoInt(Edit4.Text);

y := StrtoInt(Edit1.Text) \* x\*x + StrtoInt(Edit2.Text) \* x + StrtoInt(Edit3.Text);

x := x + 0.5;

Image1.Canvas.pen.Color := clwhite;

Image1.Canvas.LineTo(212 + Trunc(x\*10), 236 - Trunc(y\*10));

while( x <= StrtoInt(Edit5.Text) ) do

begin

y := StrtoInt(Edit1.Text) \* x\*x + StrtoInt(Edit2.Text) \* x + StrtoInt(Edit3.Text);

x := x + 0.5;

Image1.Canvas.pen.Color := clred;

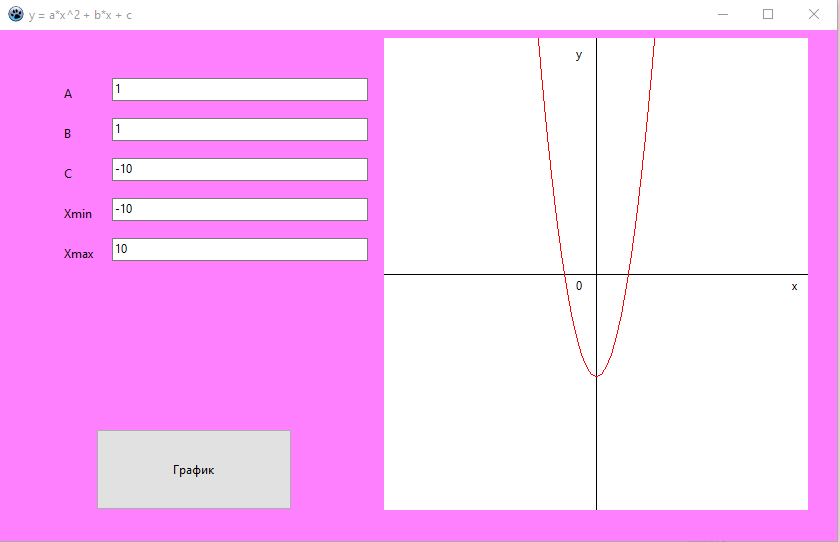
Image1.Canvas.LineTo(212 + Trunc(x\*10), 236 - Trunc(y\*10));

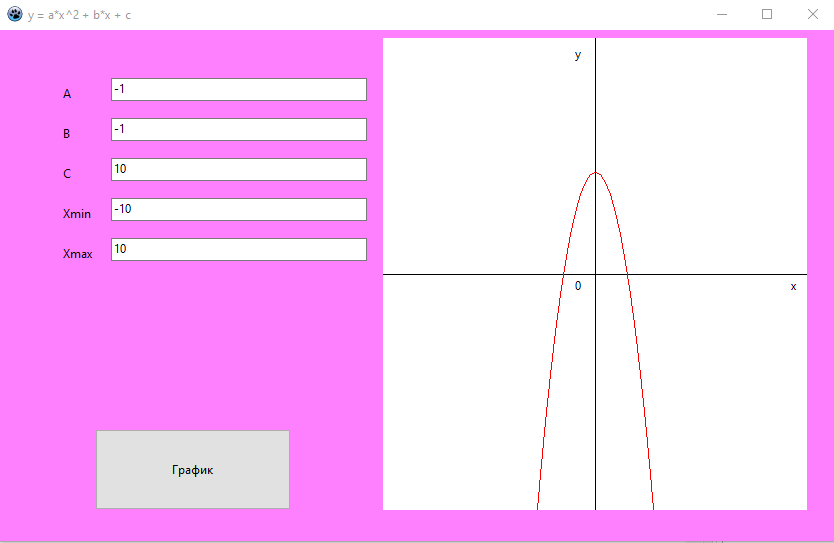
end;

end;

end.

**Скриншот программы:**





***Лабораторная работа 4***

## Тема: Движение тела брошенного под углом к горизонту

**Постановка задачи:** Изобразить местонахождение тела, брошенного с земли с начальной скорость U0 под углом к горизонту α, через равные интервалы времени ( N – число равных временных интервалов)

**Код программы:**

unit Unit1;

{$mode objfpc}{$H+}

interface

uses

Classes, SysUtils, FileUtil, Forms, Controls, Graphics, Dialogs, ExtCtrls,

StdCtrls;

type

{ TForm1 }

TForm1 = class(TForm)

Button1: TButton;

Edit1: TEdit;

Edit2: TEdit;

Edit3: TEdit;

Image1: TImage;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Label4: TLabel;

Label5: TLabel;

Label6: TLabel;

procedure Button1Click(Sender: TObject);

private

{ private declarations }

public

{ public declarations }

end;

var

Form1: TForm1;

v, a, dt, x, y, t: real;

implementation

{$R \*.lfm}

{ TForm1 }

procedure TForm1.Button1Click(Sender: TObject);

begin

Image1.Canvas.pen.Color := clblack;

Image1.Canvas.LineTo(0, 384);

Image1.Canvas.pen.Color := clred;

v := StrtoFloat(Edit1.Text);

a := StrtoFloat(Edit2.Text);

dt := StrtoFloat(Edit3.Text);

t := dt;

x := 0;

y := 384;

while (y <= 384) do

begin

x := v\*t\*((cos(a\*pi / 180)));

y := 384 - (v \* t \* ((sin(a\*pi / 180)))-( (10 \* t\*t) /2));

t := t + dt;

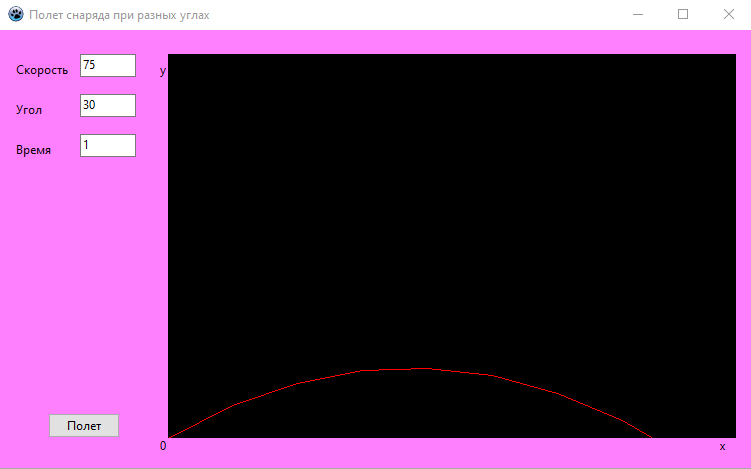
Image1.Canvas.LineTo(Trunc(x), Trunc(y));

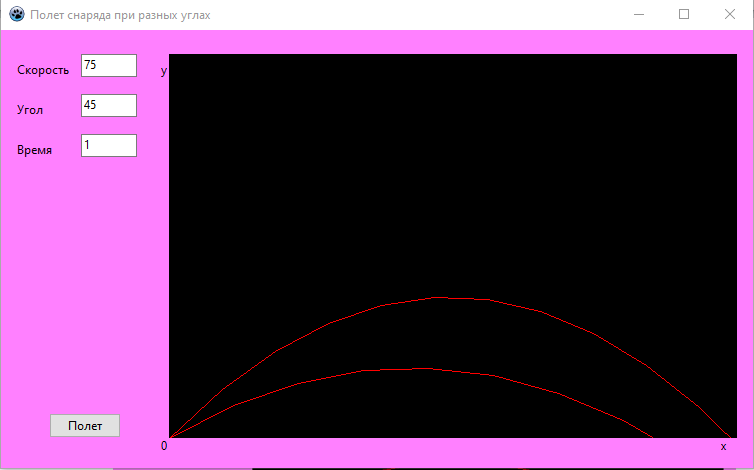
end;

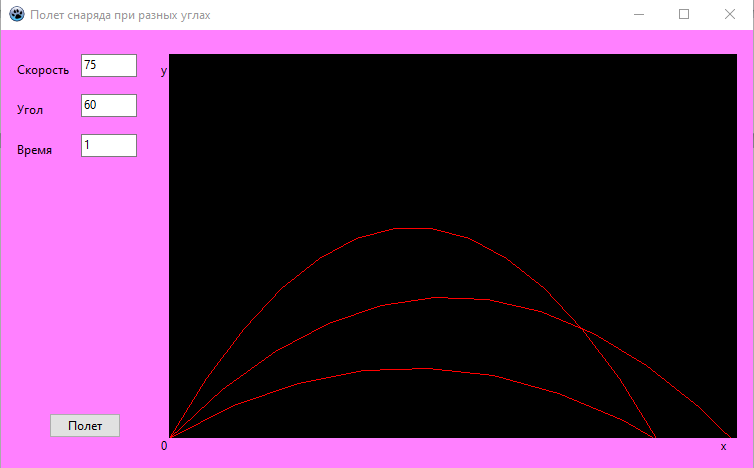
end;

end.

**Скриншот программы:**







***Лабораторная работа 5***

## Тема: Построение правильного n-угольника.

**Постановка задачи:** Построить правильный n -угольник, количество углов которого задано и вводиться через пользовательский интерфейс.

**Код программы:**

unit Unit1;

{$mode objfpc}{$H+}

interface

uses

Classes, SysUtils, FileUtil, Forms, Controls, Graphics, Dialogs, ExtCtrls,

StdCtrls;

type

{ TForm1 }

TForm1 = class(TForm)

Button1: TButton;

Edit1: TEdit;

Image1: TImage;

Label1: TLabel;

procedure Button1Click(Sender: TObject);

private

{ private declarations }

public

{ public declarations }

end;

var

Form1: TForm1;

n, i : integer;

d, dg : real;

implementation

{$R \*.lfm}

{ TForm1 }

procedure TForm1.Button1Click(Sender: TObject);

begin

Image1.Canvas.pen.Color := clblack;

Image1.Canvas.LineTo(150, 50);

Image1.Canvas.pen.Color := clred;

n := StrtoInt(Edit1.Text);

d := 360 / n ;

dg := d + 90;

for i := 1 to n - 1 do

begin

Image1.Canvas.LineTo((Trunc((cos(dg\*pi / 180)) \* 100) + 150),(-(Trunc((sin(dg\*pi/180))\*100)))+ 150);

dg := dg + d;

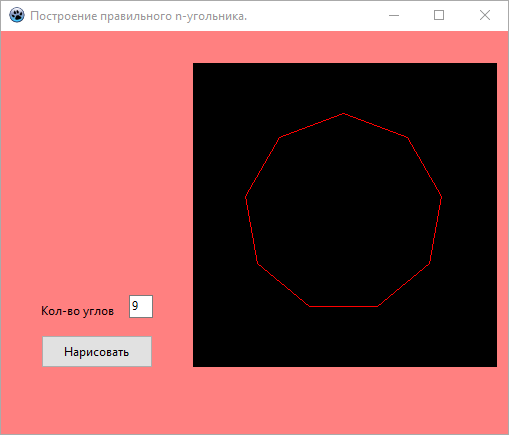
end;

Image1.Canvas.LineTo(150,50);

end;

end.

**Скриншот программы:**



***Лабораторная работа 6***

## Тема: Преобразование объекта на плоскости.

**Постановка задачи:**

### Построить произвольный объект в первой четверти системы координат (например, флажок. кол-во вершин объекта должно быть на меньше 7) и организовать следующие преобразования объекта: - общее преобразование (изменение масштаба, симметричное отражение относительно осей, сдвиг), - вращение объекта относительно начала координат - комбинированное преобразование (преобразование общего вида + вращение на произвольный угол относительно начала координат)

Матрица общего преобразования и угол поворота должны вводиться через пользовательский интерфейс, координаты вершин объекта можно указать явно в коде программы

**Код программы:**

unit Unit1;

{$mode objfpc}{$H+}

interface

uses

Classes, SysUtils, FileUtil, Forms, Controls, Graphics, Dialogs, ExtCtrls,

StdCtrls, Menus;

type

{ TForm1 }

TForm1 = class(TForm)

Button1: TButton;

CheckBox1: TCheckBox;

CheckBox2: TCheckBox;

Edit1: TEdit;

Image1: TImage;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Label4: TLabel;

RadioButton1: TRadioButton;

RadioButton2: TRadioButton;

procedure Button1Click(Sender: TObject);

private

{ private declarations }

public

{ public declarations }

end;

var

Form1: TForm1;

a : array[1..7,1..2] of integer;

t : array[1..2,1..2] of real;

i, xi, yi : integer;

q, cosq, sinq : real;

implementation

{$R \*.lfm}

{ TForm1 }

procedure TForm1.Button1Click(Sender: TObject);

begin

a[1][1]:= 65;

a[1][2]:=40;

a[2][1]:= 65;

a[2][2]:=75;

a[3][1]:= 55;

a[3][2]:=65;

a[4][1]:= 50;

a[4][2]:=75;

a[5][1]:= 45;

a[5][2]:=65;

a[6][1]:= 40;

a[6][2]:=75;

a[7][1]:= 40;

a[7][2]:=40;

q:=StrtoFloat(Edit1.Text) - 90;

cosq:=(cos(q\*pi/180));

sinq:=(sin(q\*pi/180));

for i:=1 to 7 do

begin

xi:= a[i][1]; yi:=a[i][2];

a[i][1]:=Trunc(xi\*cosq)-Trunc(yi\*sinq)+152; a[i][2]:=Trunc(xi\*sinq)+Trunc(yi\*cosq)+152;

end;

if(CheckBox1.Checked) then

begin

for i:=1 to 7 do

begin

a[i][2]:= 304 - a[i][2];

end;

end;

if(CheckBox2.Checked) then

begin

for i := 1 to 7 do

begin

a[i][1] := 304 - a[i][1];

end;

end;

Image1.Canvas.Brush.Color:= clwhite;

Image1.Canvas.Rectangle (304, 304, 0, 0);

Image1.Canvas.pen.Color := clwhite;

Image1.Canvas.LineTo(0,152);

Image1.Canvas.pen.Color := clblack;

Image1.Canvas.LineTo(304,152);

Image1.Canvas.pen.Color := clwhite;

Image1.Canvas.LineTo(152,304);

Image1.Canvas.pen.Color := clblack;

Image1.Canvas.LineTo(152,0);

Image1.Canvas.pen.Color := clwhite;

Image1.Canvas.LineTo(a[1][1],a[1][2]);

Image1.Canvas.pen.Color := clgreen;

Image1.Canvas.LineTo(a[2][1],a[2][2]);

Image1.Canvas.LineTo(a[3][1],a[3][2]);

Image1.Canvas.LineTo(a[4][1],a[4][2]);

Image1.Canvas.LineTo(a[5][1],a[5][2]);

Image1.Canvas.LineTo(a[6][1],a[6][2]);

Image1.Canvas.LineTo(a[7][1],a[7][2]);

Image1.Canvas.LineTo(a[1][1],a[1][2]);

end;

end.

**Скриншот программы:**

