

\*Примечание ко второму заданию: построение графиков в Delphi 7 с использованием Canvas. должны быть: оси, координатные сети, x, y, при любых а,b,c (они должны задаваться)- параметры

**Задача №2.
Код программы.**

unit Unit1;

interface

uses

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

 Dialogs, StdCtrls;

type

 TForm1 = class(TForm)

 Label1: TLabel;

 Edit1: TEdit;

 Label2: TLabel;

 Edit2: TEdit;

 Label3: TLabel;

 Edit3: TEdit;

 Button1: TButton;

 Memo1: TMemo;

 procedure Button1Click(Sender: TObject);

 private

 { Private declarations }

 a, b, c: Real;

 function f(x: Real): Real;

 procedure Calc;

 public

 { Public declarations }

 **end**;

var

 Form1: TForm1;

implementation

{$R \*.dfm}

// Возвращает значение функции в точке x

function TForm1.f(x: Real): Real;

begin

 Result := (Sqr(x) - a) / (x + b) + c;

**end**;

// Вычисляет корни уравнения

procedure TForm1.Calc;

const

 x1 = -50; // Границы отрезка для поиска корней

 x2 = 50;

 h = (x2 - x1) / 500; // Шаг

 e = 0.001; // Точность вычислений

var

 a, b: Real; // Границы подотрезка для поиска корней

 x: Real;

begin

 Memo1.Clear;

 a := x1; // Границы подотрезка

 b := a + h;

 while b <= x2 do begin // Цикл по всем подотрезкам

 if f(a) \* f(b) < 0 then begin // Если на этом подотрезке корень есть

 repeat // Ищем корень

 x := (a + b) / 2; // Делим подотрезок пополам

 if f(a) \* f(x) < 0 then

 b := x

 else

 a := x;

 until Abs(b - a) <= e; // Ищем корень до достижения заданной точности

 // Точку разрыва не выводим

 if not ((x >= -Form1.b - 2 \* e) and (x <= -Form1.b + 2 \* e)) then

 Memo1.Lines.Add('x = ' + FloatToStrF(x, ffFixed, 5, 3));

 **end**;

 a := b;

 b := b + h;

 **end**;

**end**;

// Рисует график

procedure TForm1.Button1Click(Sender: TObject);

const

 x0 = 450; // Точка центра координат

 y0 = 350;

var

 x, y: Integer;

 \_x, \_y: Integer;

 x1, y1: Real;

 g: Boolean;

begin

 try

 a := StrToFloat(Edit1.Text); // Вводим a, b, c

 b := StrToFloat(Edit2.Text);

 c := StrToFloat(Edit3.Text);

 except

 ShowMessage('Ошибка при вводе чисел');

 Exit;

 **end**;

 Canvas.FillRect(Rect(0, 120, ClientWidth, ClientHeight)); // Очищаем форму

 Canvas.Pen.Color := clGreen;

 x := x0 - 400;

 \_x := -20;

 while x <= x0 + 400 do begin // Рисуем сетку

 Canvas.MoveTo(x, y0 - 200);

 Canvas.LineTo(x, y0 + 200);

 Canvas.TextOut(x - 5, y0 + 200 + 5, IntToStr(\_x)); // Выводим число на форму

 Inc(x, 20);

 Inc(\_x);

 **end**;

 y := y0 - 200;

 \_y := 200;

 while y <= y0 + 200 do begin // Рисуем сетку

 Canvas.MoveTo(x0 - 400, y);

 Canvas.LineTo(x0 + 400, y);

 Canvas.TextOut(x0 - 400 - 25, y - 5, IntToStr(\_y)); // Выводим число на форму

 Inc(y, 20);

 Dec(\_y, 20);

 **end**;

 Canvas.Pen.Color := clBlue;

 Canvas.MoveTo(x0, y0 - 200 - 20); // Рисуем оси

 Canvas.LineTo(x0, y0 + 200);

 Canvas.MoveTo(x0 - 400, y0);

 Canvas.LineTo(x0 + 400 + 20, y0);

 Canvas.TextOut(x0 + 400 + 25, y0 - 10, 'X');

 Canvas.TextOut(x0 + 7, y0 - 200 - 25, 'Y');

 Canvas.Pen.Color := clRed;

 x1 := -20;

 x := x0 - 400;

 g := False;

 while x1 <= 20 do begin // Рисуем график

 y1 := f(x1);

 if (y1 >= -200) and (y1 <= 200) then begin

 if not g then begin

 Canvas.MoveTo(x, y0 - Round(y1));

 g := True;

 **end**

 else

 Canvas.LineTo(x, y0 - Round(y1));

 **end**

 else

 g := False;

 x1 := x1 + 0.05;

 Inc(x);

 **end**;

 Calc;

**end**;

**end**.

**Выполнение:**

1. Задаем параметры a,b,c.
2. Нажимаем кнопку «Построить»

*Программа построит график, а также вычислит корни методом половинного деления*.

