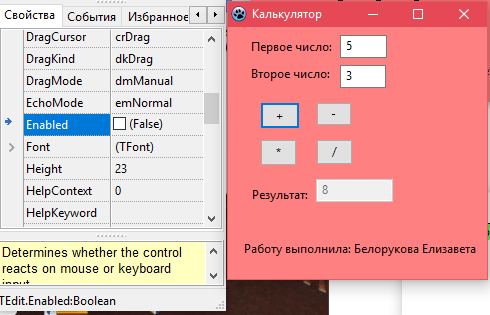
**Самостоятельная работа №1**

**Цель:**Знакомство с интегрированной средой разработки программного обеспечения Lazarus. Создание нового проекта. Знакомство с объектами формы Button, TextBox, Label, PictureBox.

**Исправленные ошибки:**

1. Составлена математическая модель
2. В кнопку результата (Edit 3) нельзя вводить числа в рабочем приложении при помощи свойства “Enabled”



**Задание №1.**

**Задание:** Заполнить таблицу «Способы визуализации»

Способы визуализации

|  |  |  |
| --- | --- | --- |
| Вид графики | Достоинства | Недостатки |
| Векторная графика | Минимальное количество информации передаётся намного меньшему размеру файла (размер не зависит от величины объекта).  При увеличении или уменьшении объектов толщина линий может быть постоянной.  Параметры объектов хранятся и могут быть изменены. | не каждый объект может быть легко изображен в векторном виде  количество памяти и времени на отображение зависит от числа объектов и их сложности  Текст (в компьютерных шрифтах, таких как TrueType, каждая буква создаётся из кривых Безье) |
| Растровая графика | Основным достоинством растровой графики принято считать то, что она обеспечивает более реалистичную передачу градаций цветов и полутонов, а также более высокую детализацию изображения, поэтому графические файлы в этом формате являются оптимальным средством представления тоновых изображений, таких как отсканированные фотографии. Еще один плюс растровой графики состоит в том, что растровое изображение проще вывести на принтер, который в своей работе использует принцип «точка на каждый элемент картинки». | При всех своих плюсах растровые изображения, использующие большую битовую глубину (16 или 24 бит на пиксель), занимают очень много места. К тому же для того, чтобы редактировать высококачественные фотографии, вам потребуется достаточно мощный компьютер с большим количеством оперативной памяти. Еще один минус состоит в том, что для растрового графического файла всегда используется фиксированное количество пикселей, поэтому любое изменение изображения (увеличение, поворот и так далее) приводит к неминуемому искажению картинки и границы объектов получаются неровными. |
| Фрактальная графка | Малый размер исполняемого файла при большом изображении.  Бесконечная масштабируемость и увеличение сложности картинки.  Незаменимость в построении сложных фигур, состоящих из однотипных элементов (облака, вода и т.д.).  Относительная легкость в создании сложных композиций.  Фотореалистичность. | Все вычисления делаются компьютером, чем сложнее изображение, тем больше загруженность ЦП и ОЗУ.  Неосвоенность технологии.  Плохое распространение и поддержка различными системами.  Небольшой спектр создания объектов изображений.  Ограниченность материнских математических фигур. |

**Задание №2**

**Задание:** Написать код по созданию калькулятора. Пользователю предлагается ввести два любых числа и произвести математические операции: сложение, вычитание, умножение и деление. Результат действия отображается в специальном окне вывода.

**Математическая модель:**

В операции «+»: c=a+b

В операции “- “: с=a-b

В операции “\*”: с=a\*b

В операции “/”: c=a/b

**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Имя переменной** | **Тип** | **Смысл** |
| a | Int(Real в делении) | Первое число(Edit1) |
| b | Int(Real в делении) | Второе число(Edit2) |
| c | Int(Double в делении) | Результат(Edit3) |

**Код программы:**

unit Unit1;

{$mode objfpc}{$H+}

interface

uses

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

type

{ TForm1 }

TForm1 = class(TForm)

Button1: TButton;

Button2: TButton;

Button3: TButton;

Button4: TButton;

Edit1: TEdit;

Edit2: TEdit;

Edit3: TEdit;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Label4: TLabel;

procedure Button1Click(Sender: TObject);

procedure Button2Click(Sender: TObject);

procedure Button3Click(Sender: TObject);

procedure Button4Click(Sender: TObject);

private

{ private declarations }

public

{ public declarations }

end;

var

Form1: TForm1;

implementation

{$R \*.lfm}

{ TForm1 }

procedure TForm1.Button3Click(Sender: TObject);

var

a,b,c: Integer;

begin

a:= StrToInt(Edit1.Text);

b:= StrToInt(Edit2.Text);

c:= a\*b;

Edit3.Text:=InttoStr(c);

end;

procedure TForm1.Button1Click(Sender: TObject);

var

a,b,c: Integer;

begin

a:= StrToInt(Edit1.Text);

b:= StrToInt(Edit2.Text);

c:= a+b;

Edit3.Text:=InttoStr(c);

end;

procedure TForm1.Button2Click(Sender: TObject);

var

a,b,c: Integer;

begin

a:= StrToInt(Edit1.Text);

b:= StrToInt(Edit2.Text);

c:= a-b;

Edit3.Text:=InttoStr(c);

end;

procedure TForm1.Button4Click(Sender: TObject);

var

a,b: Real;

c: Double;

begin

a:= StrToFloat(Edit1.Text);

b:= StrToFloat(Edit2.Text);

c:= a/b;

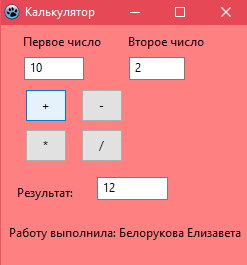
Edit3.Text:=FloattoStr(c);

end;

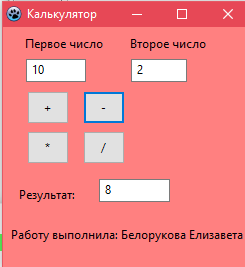
end.

**Результат работы программы:**

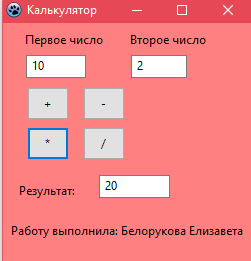
1. Сложение:



1. Вычитание:



1. Умножение:



1. Деление:

