**Самостоятельная работа № 1.**

**"Основные понятия компьютерной графики"**

1. Постановка задачи

Заполнить таблицу «Способы визуализации»:

|  |  |  |
| --- | --- | --- |
| Вид графики | Достоинства | Недостатки |
| Векторная графика | * Масштабирование размеров без потери качества изображения. * Масштабированные изображения не увеличиваются в весе ни на один байт. * Во время масштабирования качество, резкость, четкость и цветовые оттенки изображений не страдают. * Вес изображения в векторном формате в разы меньше веса изображения в растровом формате. * При конвертации изображения из векторного формата в растровый, не возникает никакой сложности. * Толщина линий при изменение масштаба (увеличение или уменьшение) объектов может не изменяться. | * В векторной графике можно изобразить далеко не каждый объект. Объем памяти и интервал времени на отображение векторной графики зависит от количества объектов и их сложности. * После преобразование из растрового изображения в векторное, обычно качество векторного изображения не высокое. |
| Растровая графика | * Растровая графика предоставляет возможность создавать любые изображения не обращая внимание на сложность их исполнения в отличие от векторной графики, которая неспособна предать хорошо переход цветов от одного к другому. * Широкий спектр применения – растровая графика на сегодняшний день нашла широкое применение в различных областях, от мелких изображений (иконок) до крупных (плакатов). * Очень высокая скорость обработки изображений различной сложности, при условие что нет необходимости в их масштабирование. * Представление растровой графики является естественным для большинства устройств и техники ввода-вывода графики. | * Большой размер файлов с простыми растровыми изображениями. * Невозможно увеличение изображения в масштабе без потери качества. * Вывод изображения при печати на плоттер является затруднительным. * При хорошем качестве изображения требуются значительный объем дискового пространства для хранения файлов. * Сложность преобразования растрового изображения в векторное. |
| Фрактальная графика | * Отличное качество изображения * Поддерживаются случайно-порождённые фракталы * Позволяет строить фракталы в больших разрешениях. | Необходимо знание языка программирования (скрипты) |

2. Постановка задачи

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

с=a+b

с=a-b

с=a/b

с=a\*b

Список идентификаторов

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| a | Первое вводимое пользователем число | При сложении, вычитании и умножении integer, при делении double |
| b | Второе вводимое пользователем число | При сложении, вычитании и умножении integer, при делении double |
| c | Результат вычислений | При сложении, вычитании и умножении integer, при делении double |

Код программы

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.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.Button4Click(Sender: TObject);

var

a,b,c:double;

begin

a:=StrTofloat(Edit1.Text);

b:=StrTofloat(Edit2.Text);

c:=a/b;

Edit3.Text:=floattoStr(c);

end;

Результаты выполненной работы

Была исправлена кнопка с результатом с помощью свойства Enabled в Edit3:TEdit

 