

Занятие 2. Графика

Приступая к решению задач, необходимо вспомнить:

Программа может выводить графику на поверхность объекта (формы или компонента image), которой соответствует свойство Canvas.

Для того чтобы на поверхности объекта появился графический элемент (линия, окружность, прямоугольник и т. д.) или картинка, необходимо применить к свойству Canvas этого объекта соответствующий метод.

Цвет, стиль и толщину линий, вычерчиваемых методами Line, Ellipse, Rectangle и т. д., определяет СВОЙСТВО Pen объекта Canvas.

Цвет закраски внутренних областей геометрических фигур, вычерчиваемых методами Line, Ellipse, Rectangle И Т. Д., определяет свойство Brush объекта Canvas.

Характеристики шрифта текста, выводимого методом Textout, определяет свойство Font объекта canvas.

Основную работу по выводу графики на поверхность формы должна выполнять функция обработки события onPaint.

ЗАДАНИЕ 1. Напишите программу, которая на поверхности формы рисует олимпийский флаг (рис. 2.1).

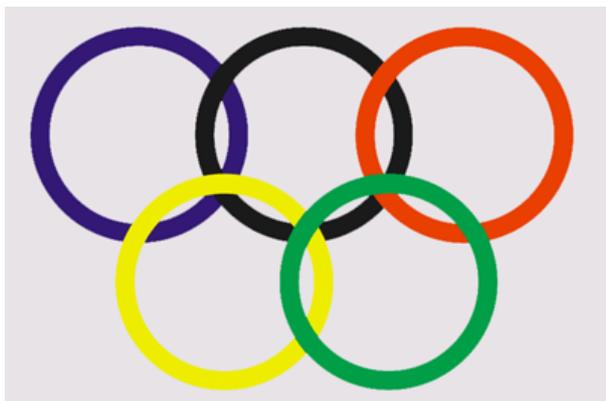


Рис. 2.1. Олимпийский флаг

```
// процедура рисует олимпийский флаг
procedure TForm1.FormPaint(Sender: TObject);
begin
  with Canvas do
    begin
      // полотнище
      Canvas.Pen.Width := 1;
      Canvas.Pen.Color := clBlack;
      Canvas.Brush.Color := clCream;
      Rectangle(30,30,150,115);
      // кольца
      Pen.Width := 2;
      Brush.Style := bsClear; // область внутри круга не закрашивать
      Pen.Color := clBlue;
      Ellipse(40,40,80,80);
      Pen.Color := clBlack;
      Ellipse(70,40,110,80);
      Pen.Color := clRed;
      Ellipse(100,40,140,80);
      Pen.Color := clYellow;
```

```

Ellipse(55,65,95,105);
Pen.Color := clGreen;
Ellipse(85,65,125,105);
end;
end;

```

ЗАДАНИЕ 2. Напишите программу, которая на поверхность формы выводит изображение оцифрованной координатной сетки (рис.2.2).



Рис. 2.2. Координатная сетка

```

// обработка события OnPaint
procedure TForm1.FormPaint(Sender: TObject);
var
x0,y0:integer; // координаты начала координатных осей
dx,dy:integer; // шаг координатной сетки (в пикселях)
h,w:integer; // высота и ширина области вывода координатной
// сетки
x,y:integer; lx,ly:real; // метки (оцифровка) линий сетки по X и Y
dlx,dly:real; // шаг меток (оцифровки) линий сетки по X и Y
cross:integer; // счетчик не оцифрованных линий сетки
dcross:integer; // количество не оцифрованных линий
// между оцифрованными
begin
x0:=30; y0:=220; // оси начинаются в точке (40,250)
dx:=40; dy:=40; // шаг координатной сетки 40 пикселей
dcross:=1; // помечать линии сетки X: 1 – каждую;
// 2 – через одну;
// 3 – через две;
dlx:=0.5; // шаг меток оси X
dly:=1.0; // шаг меток оси Y, метками будут: 1,2,3 и т. д.
h:=200;
w:=300;
with form1.Canvas do
begin
cross:=dcross ;
MoveTo(x0,y0); LineTo(x0,y0-h); // ось X
MoveTo(x0,y0); LineTo(x0+w,y0); // ось Y
// засечки, сетка и оцифровка по оси X
x:=x0+dx;
lx:=dlx;
repeat

```

```

MoveTo(x,y0-3);LineTo(x,y0+3) ; // засечка
cross:=cross-1;
if cross = 0 then //оцифровка
begin
  TextOut(x-8,y0+5,FloatToStr(lx)) ;
  cross:=dcross;
end;
Pen.Style:=psDot;
MoveTo(x,y0-3);LineTo(x,y0-h); // линия сетки
Pen.Style:=psSolid;
lx:=lx+dlx;
x:=x+dx;
until (x>x0+w);
// засечки, сетка и оцифровка по оси Y
y:=y0-dy;
ly:=dly;
repeat
  MoveTo(x0-3, y);LineTo(x0+3,y) ; // засечка
  TextOut(x0-20,y,FloatToStr(ly)) // оцифровка
  Pen.Style:=psDot ;
  MoveTo(x0+3,y); LineTo(x0+w,y); // линия сетки
  Pen.Style:=psSolid;
  y:=y-dy;
  ly:=ly+dly;
until (y<y0-h);
end;
end;

```

ЗАДАНИЕ 3. Напишите программу, которая на поверхности формы вычерчивает график функции, например $y = 2 \sin x \cdot e^{\frac{x}{5}}$. Вид окна во время работы программы приведен на рис. 2.3.

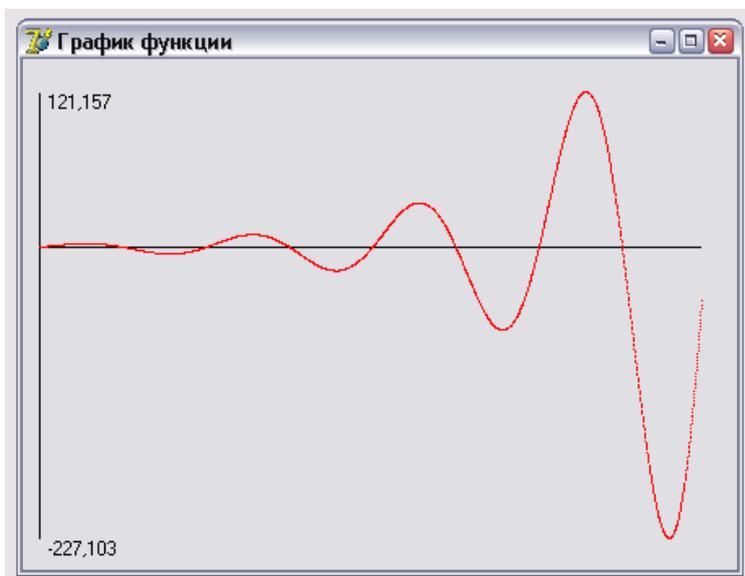


Рис. 2.3. Окно программы График функции

```

// Функция, график которой надо построить
Function f(x:real):real;
begin
  f:=2*Sin(x)*exp(x/5);
end;

```

```

// процедура строит график функции
procedure GrOfFunc;
var
x1,x2:real; // границы изменения аргумента функции
y1,y2:real; // границы изменения значения функции
x:real; // аргумент функции
y:real; // значение функции в точке x
dx:real; // приращение аргумента
l,b:integer; // левый нижний угол области вывода графика
w,h:integer; // ширина и высота области вывода графика
mx,my:real; // масштаб по осям X и Y
x0,y0:integer; // точка – начало координат
begin
// область вывода графика
l:=10; // X – координата левого нижнего угла
b:=Form1.ClientHeight-20; // Y – координата левого нижнего угла
h:=Form1.ClientHeight-40; // высота
w:=Form1.Width-40; // ширина
x1:=0; // нижняя граница диапазона аргумента
x2:=25; // верхняя граница диапазона аргумента
dx:=0.01; // шаг аргумента
// найдем максимальное и минимальное значения
// функции на отрезке [x1,x2]
y1:=f(x1); // минимум
y2:=f(x2); //максимум
x:=x1;
repeat
  y := f(x);
  if y < y1 then y1:=y;
  if y > y2 then y2:=y;
  x:=x+dx;
until (x>=x2);
// вычислим масштаб
my:=h/abs(y2-y1); // масштаб по оси Y
mx:=w/abs(x2-x1); // масштаб по оси X
x0:=1;
y0:=b-Abs(Round(y1*my));
with Form1.Canvas do
  begin
    // оси
    MoveTo(l,b);LineTo(l,b-h);
    MoveTo(x0,y0);LineTo(x0+w, y0) ;
    TextOut (l+5,b-h,FloatToStrF(y2,ffGeneral,6,3) ) ;
    TextOut(l+5,b,FloatToStrF(y1,ffGeneral, 6,3) ) ;
    // построение графика
    x:=x1;
    repeat
      y:=f(x);
      Pixels[x0+Round(x*mx),y0-Round(y*my)]:=clRed;
      x:=x+dx;
    until (x>=x2);
  end;
end;

// обработка события onPaint
procedure TForm1.FormPaint(Sender: TObject);
begin

```

```
GrOfFunc;  
end;  
  
// если изменился размер окна программы  
procedure TForm1.FormResize(Sender: TObject);  
begin  
// очистить форму  
Form1.Canvas.FillRect(Rect(0,0,ClientWidth,ClientHeight));  
// построить график  
GrOfFunc;  
end;
```

ЗАДАНИЕ 4. Используя результаты выполнения предыдущих заданий, напишите программу, которая на поверхности формы вычерчивает график функции на фоне оцифрованной координатной сетки.