И. А. Буяковская

ПОСТРОЕНИЕ ГРАФИКОВ ФУНКЦИЙ В БЕСПЛАТНЫХ СРЕДАХ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО ПРОГРАММИРОВАНИЯ

Раздел программирования, в котором рассматриваются принципы построения графиков функций включает три категории задач на построение графика функции, заданного в явном виде, в параметрическом виде и в полярных координатах. Общие подходы, применяемые при решении такого рода задач сходно реализуются в средах объектно-ориентированного программирования Lazarus и Gambas. Отличительные особенности программного кода, в данном случае, обусловлены синтаксисом и компонентами, применяемыми при разработке оконного приложения.

В среде Lazarus используются компоненты, обладающие свойством Canvas, а сам компонент рассматривается как прямоугольная сетка, состоящая из отдельных точек. Свойством Canvas обладают такие компоненты как: форма (класс Tform), область рисования (класс TpaintBox), растровая картинка (класс Timage), панель (класс Tpanel).

Рассмотрим пример построения графика функции непосредственно на форме в среде Lazarus, заданного в параметрическом виде, где

$$\begin{cases} x = r\square(t - \sin t) \\ y = r\square(1 - \cos t) \end{cases}$$

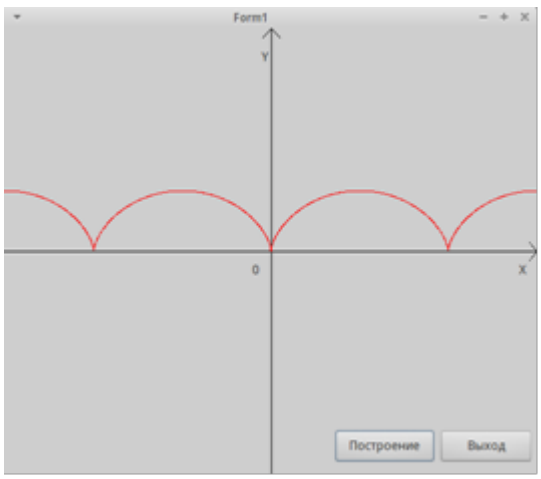

Рис. 1. График функции в среде Lazarus

На форме разместим две кнопки класса Tbutton из вкладки Standard. Построение графика будет осуществляться по нажатию на кнопку (“Построение”). Программный код которой приведем ниже:

```
procedure TForm1.Button1Click(Sender: TObject);

var r, d, x01, y01 :integer;

    t,x,y:real;
```

```
begin

  r:=30; // Задаем значение переменной, отвечающей за масштабирование графика функции.

   d:=10; //  Задаем длину направляющих стрелок декартовой системы координат.

  x01:=Form1.Width div 2; // Определим начало координат.

  y01:=Form1.Height div 2;

  Form1.Canvas.Line(0,y01,x01*2,y01); // Построим ось абсцисс.

  Form1.Canvas.Line(x01,0,x01,y01*2); // Построим ось ординат.

   Form1.Canvas.Line(x01,0,x01+d,d);  //Построим направляющие для оси оридинат.

  Form1.Canvas.Line(x01,0,x01-d,d);

   Form1.Canvas.Line(x01*2,y01,x01*2-d,y01+d);  //Построим направляющие для оси абсцисс.

  Form1.Canvas.Line(x01*2,y01,x01*2-d,y01-d);

  Form1.Canvas.TextOut(x01-d,d*2,'Y'); // Подпишем оси и начало координат.

  Form1.Canvas.TextOut(x01*2-d*2,y01+d,'X');

  Form1.Canvas.TextOut(x01-d*2,y01+d,'0');

  t:=-10; // Зададим начальное значение параметра.

  While t<=10  do

  begin

     x:=r*(t-sin(t));  //  Определим коорднату точки кривой, заданной в параметрическом представлении.

    y:=-r*(1-cos(t));

    t:=t+0.01; // Увеличим значение параметра с шагом 0,01.

    Form1.Canvas.Pixels[trunc(x01+x),trunc(y01+y)]:=clred; //Построим точку красного цвета с вычисленными координатами.

  end;

end;
```

Теперь рассмотрим пример построения графика функции в среде Gambas, заданного явно, где $y=\frac{(x-4)}{\sqrt{x^2+2}}$. Для этого на форме разместим компонент DrawingArea из вкладки Container.

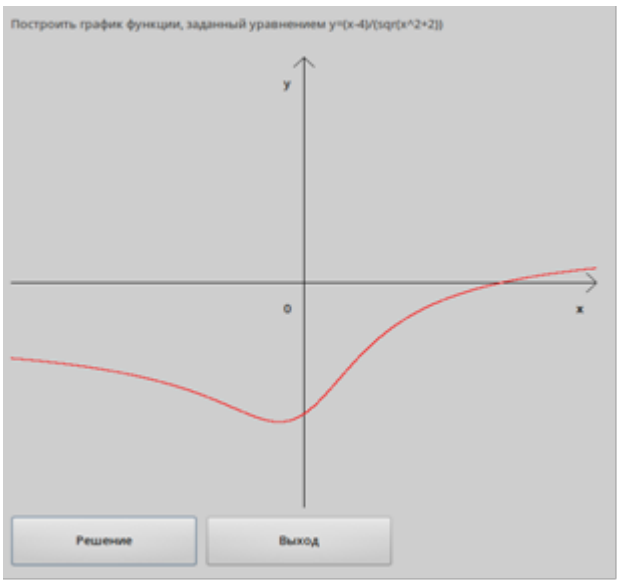

Рис. 2. График функции в среде Gambas

```
PUBLIC SUB Button1_Click()

DIM d AS Byte

DIM x0, y0, x, y, t, r AS Single

  r = 50

  d = 10

  Draw.Begin(DrawingArea1)

  x0 = Draw.Width DIV 2

  y0 = Draw.Height DIV 2

  Draw.ForeColor = Form.Background

  Draw.Line(0, y0, x0*2, y0)

  Draw.Line(x0, 0, x0, y0*2)

  Draw.Line(x0, 0, x0 - d, d)

  Draw.Line(x0, 0, x0 + d, d)

  Draw.Line(x0*2, y0, x0*2 - d, y0 - d)

  Draw.Line(x0*2, y0, x0*2 - d, y0 + d)

  Draw.Text("y", x0 - 2 * d, 2 * d)
```

```
  Draw.Text("x", x0*2 - 2 * d, y0 + 2 * d)
  Draw.Text("0", x0 - 2 * d, y0 + 2 * d)
  draw.ForeColor = Color.Red
  FOR x = -20 TO 20 STEP 0.001
    y = y0 - r * ((x - 4) / Sqr((x ^ 2 + 2)))
    Draw.Point(Round(x0 + x * r), Round(y))
  NEXT
  Draw.End
END
```