

# Построение графиков полярных кривых в Visual Basic

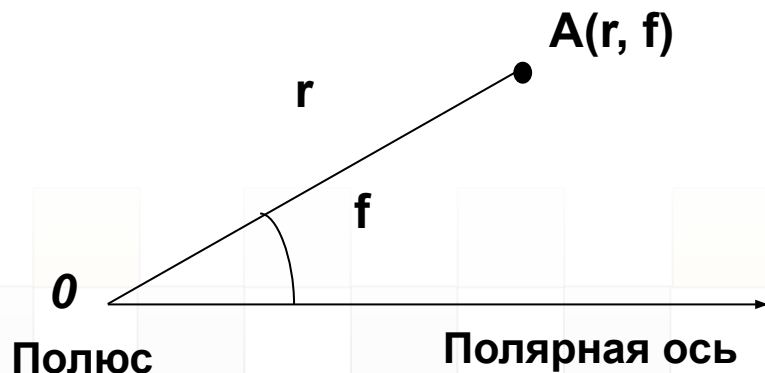
Практикум по решению задач на ЭВМ

Г.В.Гибадулина, 2012

# Понятие полярной системы

**Полярная система** координат образуется полярным полюсом  $O$  и полярной осью, проведённой из полюса в направлении слева направо.

В полярной системе любой точке соответствует единственная пара полярных координат  $A(r, f)$ , где  $r$  – полярный радиус, а  $f$  – полярный угол.

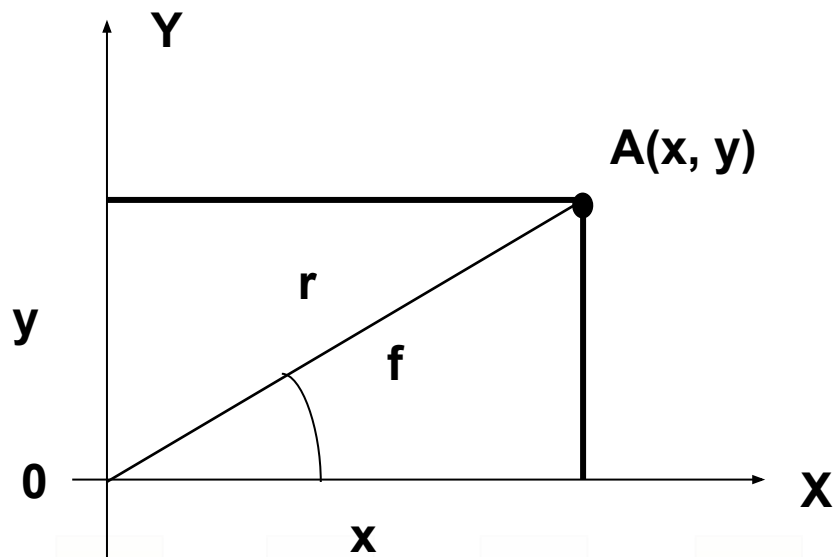


**Полярный радиус  $r$**  – отрезок, соединяющий полюс с заданной точкой  $A$ .

**Полярный угол  $f$**  – угол между полярной осью и полярным радиусом.

# Переход от полярной системы координат к декартовой

Нужно совместить полярную и декартову прямоугольную системы таким образом, чтобы начала их координат совпадали, а полярная ось совпадала с положительным направлением оси абсцисс X.



**Формулы перехода от полярных координат к декартовым (параметрические уравнения):**

$$\begin{aligned}x &= r \cdot \cos f \\ y &= r \cdot \sin f\end{aligned}$$

**Формулы обратного перехода от декартовых координат к полярным:**

$$f = \operatorname{arctg}(y/x); \quad r = \sqrt{x^2 + y^2}$$

# Построение графиков полярных кривых в Visual Basic

## Задание

Создать проект в среде Visual Basic для построения графиков полярных кривых, заданных параметрическими уравнениями: кардиоиды, логарифмической спирали, декартова листа, фигуры Лиссажу,  $k$ -лепестковой розы, эпициклоиды

# Интерфейс проекта

Form1

Построение полярных кривых

График

Выбор типа полярной кривой

- кардиоида
- логарифмическая спираль
- декартов лист
- фигура Лиссажу
- k-лепестковая роза
- эпициклоида

фигура Лиссажу

```
r = 5  
x = r * Cos(3*f)  
y = r * Sin(2*f)
```

Пуск Сброс Выход

Label3

Label1

Label2

Frame1

Массив переключателей Option1(i)

Text1

Picture1

Command1

# Свойства элементов управления

Option1(0)

Выбор типа полярной кривой

- кардиоида
- логарифмическая спираль
- декартов лист
- фигура Лиссажу
- k-лепестковая роза
- эпициклоида

Option1(5)

Элемент управления	Свойство	Значение
Frame1	Caption	Выбор типа полярной кривой
Option1(0)	Caption	кардиоида
Option1(1)	Caption	логарифмическая спираль
Option1(2)	Caption	декартов лист
Option1(3)	Caption	k-лепестковая роза
Option1(4)	Caption	фигура Лиссажу
Option1(5)	Caption	эпициклоида

# Свойства элементов управления

Элемент управления	Свойство	Значение
Label3	Caption	Построение полярных кривых
Label1	Caption	График
Label1, Label2, Label3	Alignment	2 - center
Text1	Text	
Text1	Alignment	0 - left
Picture1	FillStyle	0 - solid
Command1	Caption	Пуск

# Описание переменных

**x, y** – координаты точки

**r** – радиус; **f** - угол

**Pi** –число Пи

**a, b, q, k** – коэффициенты в параметрических уравнениях

**z** - индекс выбранного переключателя

**i** – параметр цикла для задержки рисования на экране

```
Dim z, k As Integer, f, pi, a, r, b,  
    q As Single, i As Long, x, y As  
    Single
```



# Процедура выбора типа кривой

Анализируем значение свойства Value каждого переключателя.

```
Private Sub Option1_Click(Index As Integer)
```

‘в переменной z запоминаем индекс выбранного переключателя

```
For i = 0 To 5
```

```
    If Option1(i).Value = True Then z = i
```

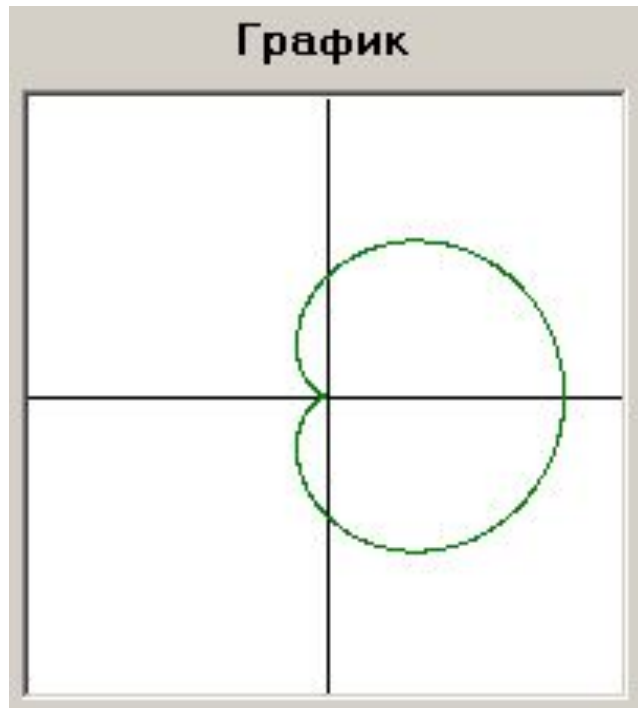
```
Next
```

```
End Sub
```

# Процедура кнопки Пуск

```
Private Sub Command1_Click()  
    'шкала  
    Picture1.Scale (-10, 10)-(10, -10)  
    'оси координат  
    Picture1.Line (-10, 0)-(10, 0),  
        QBColor(0)  
    Picture1.Line (0, -10)-(0, 10),  
        QBColor(0)  
    pi = 3.14154
```

# Кардиоида



## Параметрические уравнения

$$r = a * (1 + \text{Cos}(f))$$

$$x = r * \text{Cos}(f)$$

$$y = r * \text{Sin}(f)$$

$$f \in [0, 2*\pi]$$

# Построение графика кардиоиды

```
If z = 0 Then
  a = 4:    q = 4:    f = 0
  While f <= 2 * pi
    r = a * (1 + Cos(f))
    x = r * Cos(f)
    y = r * Sin(f)
    'построение точки
    Picture1.PSet (x, y), QBColor(2)
    'задержка рисования
    For i = 1 To 100000: Next
    f = f + 0.01
  Wend
```

# Построение графика кардиоиды

‘Вывод параметрических уравнений в текстовое поле

```
Text1.Text = Text1.Text & "r = a *  
(1 + Cos(f))" & Chr(13) & Chr(10)
```

```
Text1.Text = Text1.Text & "x = r *  
Cos(f)" & Chr(13) & Chr(10) & "y = r  
* Sin(f)"
```

```
End If
```

```
End Sub
```

# Процедура кнопки Сброс

```
Private Sub Command3_Click()  
    Picture1.Cls  
    Picture1.Line (-10, 0)-(10, 0),  
        QBColor(0)  
    Picture1.Line (0, 10)-(0, -10),  
        QBColor(0)  
    Label2.Caption = ""  
    Text1.Text = ""  
End Sub
```

# Логарифмическая спираль



## Параметрические уравнения

$$a = 0.2$$

$$b = 0.15$$

$$n = 5$$

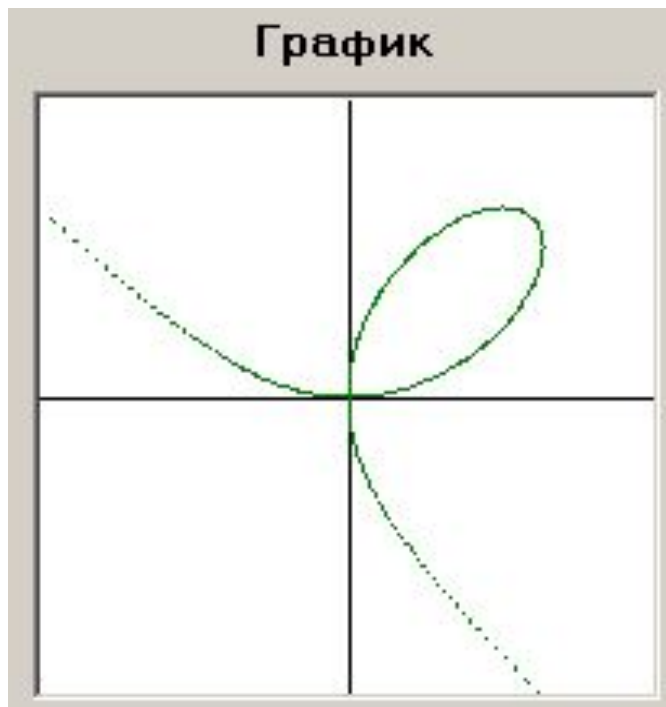
$$r = a * \text{Exp}(b * f)$$

$$x = r * \text{Cos}(f)$$

$$y = r * \text{Sin}(f)$$

$$f \in [0, 2 * \pi * n]$$

# Декартов лист



## Параметрические уравнения

$$a = 4$$

$$r = 3 * a * \cos(f) * \frac{\sin(f)}{(\cos(f) \wedge 3 + \sin(f) \wedge 3)}$$

$$x = r * \cos(f)$$

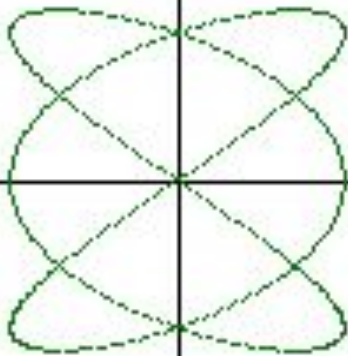
$$y = r * \sin(f)$$

$$f \in [0, 2 * \pi]$$



# Фигура Лиссажу

График



## Параметрические уравнения

$$r = 5$$

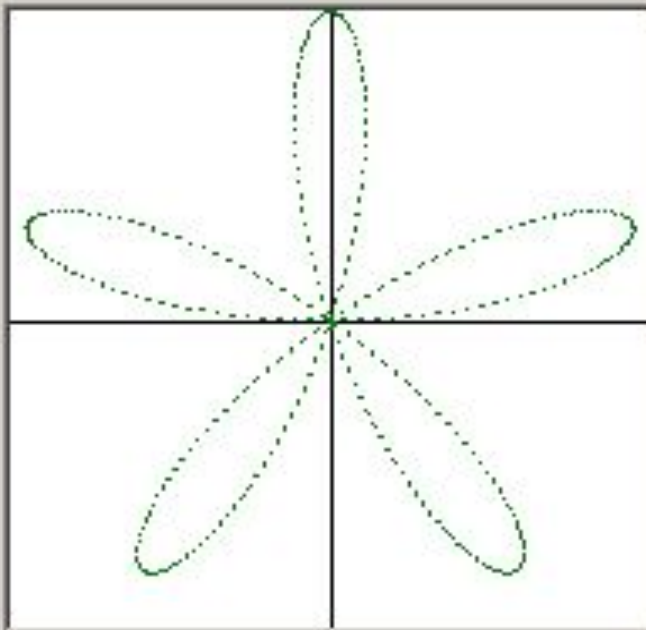
$$x = r * \text{Cos}(3 * f)$$

$$y = r * \text{Sin}(2 * f)$$

$$f \in [-\pi, \pi]$$

# К-лепестковая роза

График



## Параметрические уравнения

При чётных значениях  $k$  получается  $2*k$ -лепестковая роза, а при нечётных значениях –  $k$ -лепестковая роза.

$$r = \text{Sin}(k * f)$$

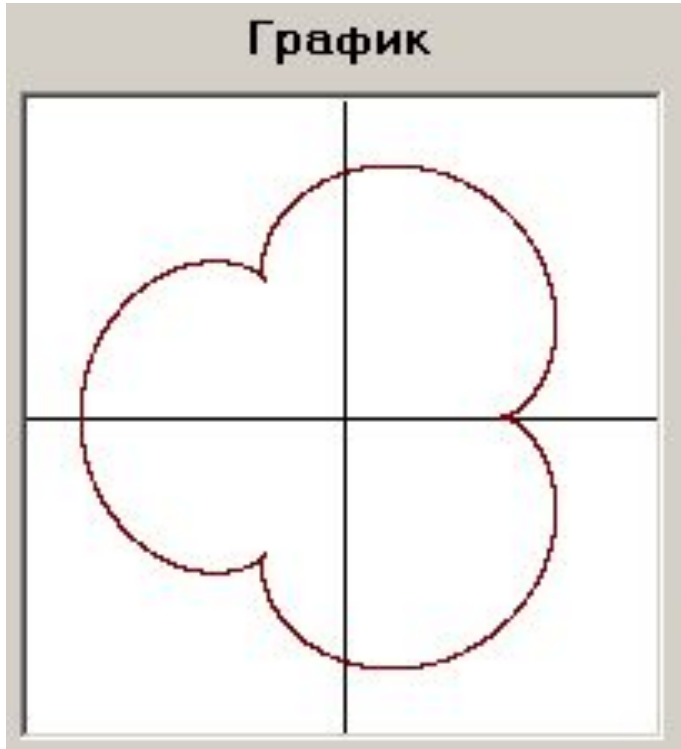
$$x = 10 * r * \text{Cos}(f)$$

$$y = 10 * r * \text{Sin}(f)$$

$$f \in [-\pi, \pi]$$

# Эпициклоида

График



## Параметрические уравнения

$$x = (a + b) * \text{Cos}(f) - a$$
$$* \text{Cos}((a + b) * f / a)$$

$$y = (a + b) * \text{Sin}(f) - a$$
$$* \text{Sin}((a + b) * f / a)$$

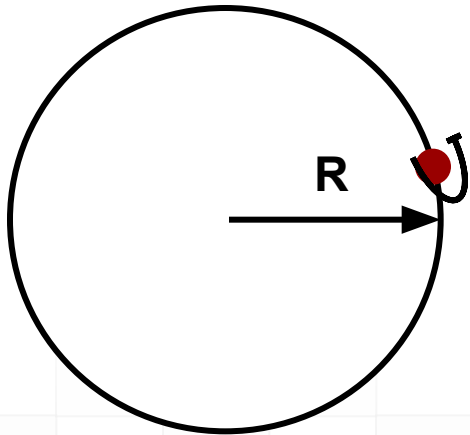
$$f \in [0, 2 * \pi]$$

Значение b вводится с клавиатуры.

Значение a примем равным  $a = b / 3$ .

# Задания для самостоятельной работы

1. Добавьте в проект кнопку Движение (Command4). Реализовать движение объекта (окружности) по траектории окружности радиуса  $R$ .



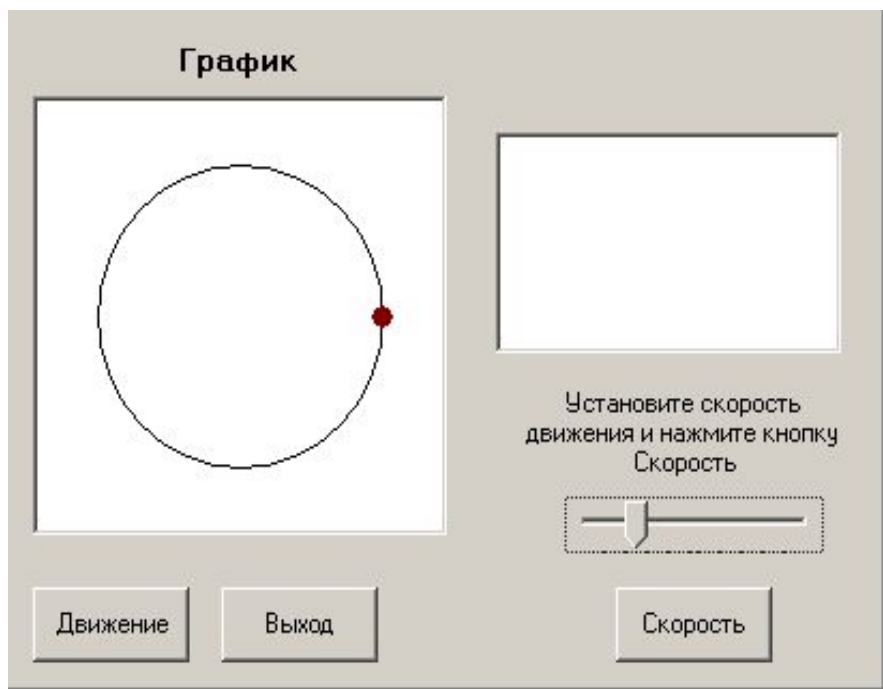
## Параметрические уравнения окружности

$$x = r * \text{Cos}(f)$$

$$y = r * \text{Sin}(f)$$

$$f \in [0, 2 * \pi]$$

# Задания для самостоятельной работы



2. Добавьте в проект элемент управления ползунков **Slider** (Проект – Компоненты – Microsoft Windows Common Controls6), для регулировки скорости движения окружности, и кнопку **Скорость** (Command5). Установите свойство ползунка Min=1, Max=20.

Slider1

Command5

# Задания для самостоятельной работы

## ‘процедура кнопки Скорость

```
Private Sub Command5_Click()  
    v = Slider1.Value  
End Sub
```

‘переменную *v* используйте в цикле для  
задержки рисования

```
For i = 1 To 100000*v : Next
```

# Процедура кнопки Движение

```
Private Sub Command4_Click() 'движение
    pi = 3.1415: r = 7
    Picture1.Scale (-10, 10)-(10, -10)
    For f = 0 To 2 * pi Step 0.01
        Picture1.FillColor = QBColor(4) 'цвет заливки
        x = r * Cos(f) 'параметрические уравнения окружности
        y = r * Sin(f)
        Picture1.Circle (x, y), 0.5, QBColor(4) 'рисуем
    окружность
    For i = 1 To 10000 * v: Next 'задержка рисования
    Picture1.FillColor = QBColor(15) 'цвет фона
    Picture1.Circle (x, y), 0.5, QBColor(15) 'стираем
    окружность цветом фона
    For i = 1 To 10000 * v: Next
Next
End Sub
```