

Программа: Число π и метод Монте Карло



Курс «Основы программирования в среде **Visual Basic**»
Преподаватель: Тюкавина Татьяна Михайловна.
МБОУ ДО Гатчинский Центр Непрерывного Образования
«Центр Информационных Технологий»

π - математическая константа, равная отношению длины окружности к длине её диаметра.

Обозначается буквой греческого алфавита π . Старое название — *лудольфово число*.

См. [ПИ – википедия](#)

Впервые обозначением этого числа греческой буквой воспользовался британский математик Джонс (1706), а общепринятым оно стало после работ [Леонарда Эйлера](#) в 1737. Это обозначение происходит от начальной буквы греческих слов *περιφέρεια* — окружность, периферия и *περίμετρος* — периметр. Очень интересно про историю этой константы и способы измерений изложены в [wikia](#) и в [Словари и энциклопедии на Академикe](#)



1. Алгоритм Бюффона для определения числа Пи

Случайные величины использовались для решения прикладных задач достаточно давно. Примером может служить способ определения числа Пи, который был предложен Бюффоном еще в 1777 году. См. [Википедия](#)

2. Для написания программы используем **Метод статистического моделирования, или метод Монте-Карло.** ([Паньгина Н.Н, Паньгин А.А. –Школа современного программирования.](#))

Методы Монте-Карло, ММК — общее название группы численных методов, основанных на получении большого числа реализаций случайного процесса, который формируется таким образом, чтобы его вероятностные характеристики совпадали с аналогичными величинами решаемой задачи. Используется для решения задач в различных областях физики, химии, математики, экономики, оптимизации, теории управления и др.

Метод получил свое название по имени города Монте-Карло в княжестве Монако, известного своими игорными заведениями, в которых публика растрчивает или увеличивает свои доходы, согласно *законам распределения случайных величин*. Дело в том, что одним из механических приборов для получения случайных величин является рулетка.

Метод Монте-Карло



Это метод статистических испытаний – численный метод, основанный на моделировании случайных величин и построении статистических оценок для искомых величин.

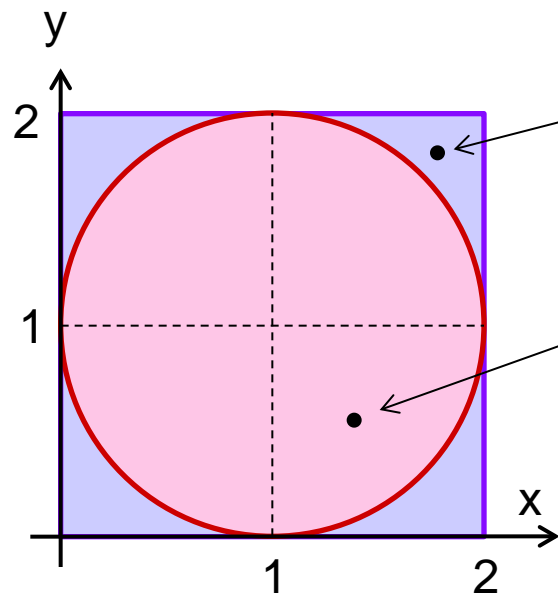
Метод статистического моделирования, или метод **Монте-Карло** используется для *моделирования игровых вероятностных ситуаций* (бросание монеты, кубика, блуждания), а также для *вычисления площадей фигур неопределенной формы*.

Для вычисления площади некоторой фигуры проведем эксперимент: поместим фигуру в квадрат и будем случайным образом бросать точки в этот квадрат. Чем больше площадь фигуры, тем чаще в нее будут попадать точки. Таким образом, можно сделать допущение: при большом числе точек, наугад выбранных **внутри квадрата**, доля точек, содержащихся **в данной фигуре**, приближенно равна **отношению площади этой фигуры к площади квадрата**.

С появлением ЭВМ этот метод несложно реализовать.

Рассмотрим имитационное моделирование на примере

Вычисление числа $\pi = 3,1415922653\dots$ методом Монте-Карло.



Точка принадлежит квадрату,
если:
 $0 \leq x \leq 2$ и $0 \leq y \leq 2$

Точка принадлежит кругу,
если:
 $(x-1)^2 + (y-1)^2 \leq 1$

Докажите это неравенство, решив
геометрическую задачу.

Эксперимент: бросание случайных точек

m – число точек, попавших в круг

n – общее число точек

Тогда:
$$\frac{S_0}{S_S} = \frac{m}{n} \quad (2)$$

Площадь круга : $S_0 = \pi r^2$

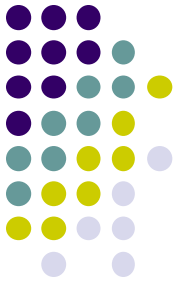
Площадь квадрата : $S_S = 4r^2$

Тогда:
$$\frac{S_0}{S_S} = \frac{\pi}{4} \quad (1)$$

Из выражения (1) и (2) следует:

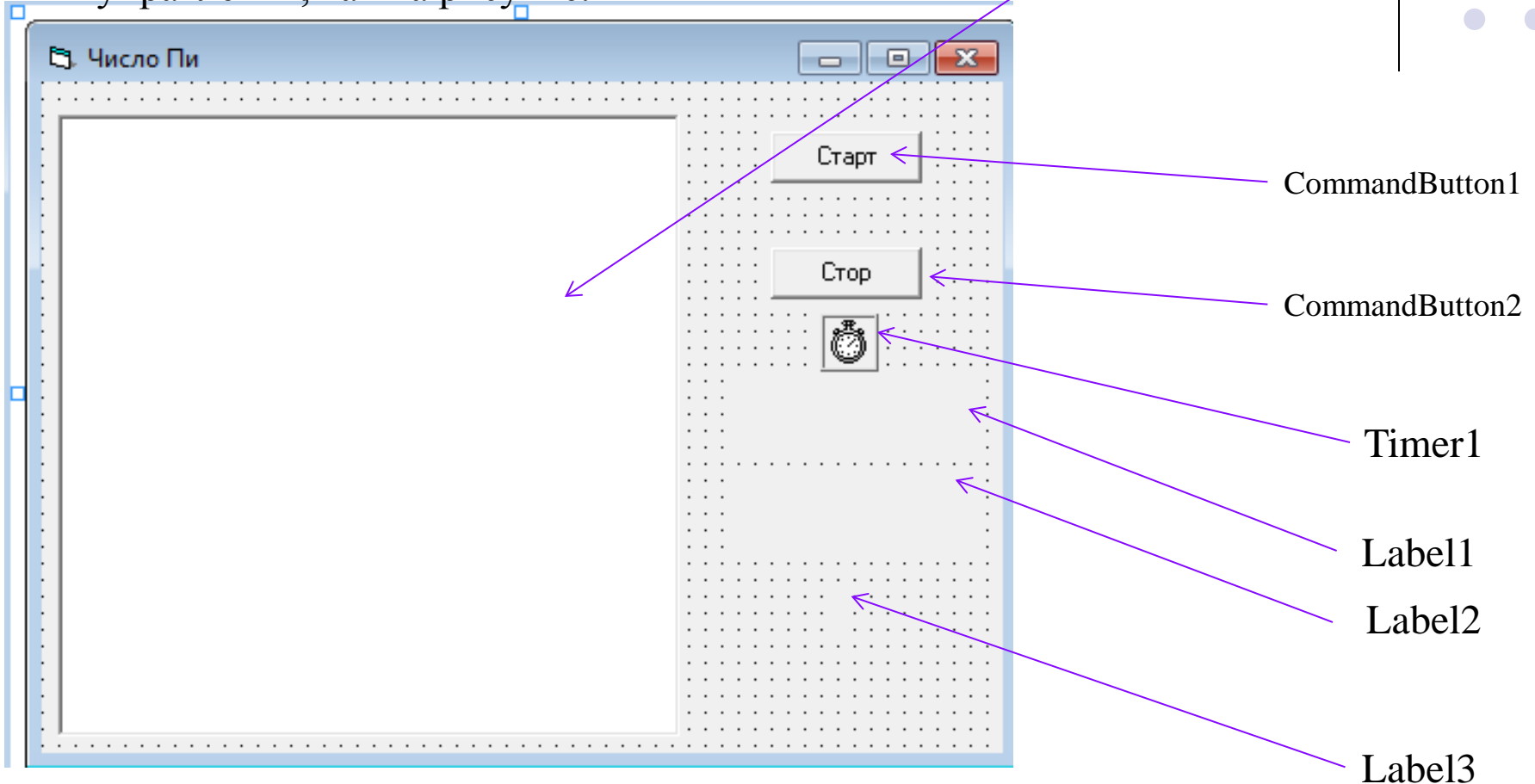
$$\pi = \frac{4 \cdot m}{n} \quad (3)$$

Программа в среде Visual Basic



I. Визуальное программирование.

1. Создаем форму и расставим на ней объекты управления, как на рисунке.



при конструировании задаем свойства:

Для меток Label1, Label2, Label3

Alignment = 2 - Center,

Autosize = True

2. На объекте PictureBox (Окно рисунка) будем рисовать квадрат и вписанный в него круг.

Задаем в окне свойств Properties значения:

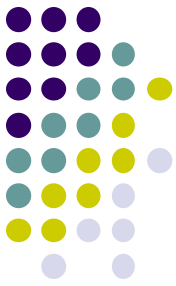
Name = Pic

Autosize = False

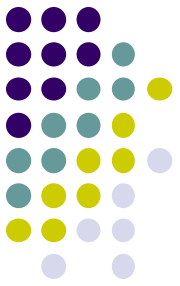
Picture = None т.е. готовую картинку не загружаем, т.к. рисовать квадрат и вписанный в него круг будем в программе, используя соответствующие графические методы.

Width и Height должны *быть равными* для рисования квадрата

3. На форму помещаем невидимый объект управления Timer1 для задания скорости «разбрасывания» точек в случайное место квадрата. При конструировании программы задаем в окне свойств Properties значения :
Interval = 1 (что означает 1 миллисекунда)
Enabled = False (т.е. в исходном состоянии часики стоят)
4. Для меток Label1, Label2, Label3 в окне свойств Properties задаем свойство Caption пустым, т.к. значения получим и выведем в программе.
Label1 – в эту метку выводим число точек **m**, попавших в круг
Label2 – в эту метку выводим общее число точек **n**, т.е. попавших в квадрат
Label3 – в эту метку выводим вычисленное значение числа Pi
5. Разместим на форме 2 командные кнопки с соответствующим свойством Caption «Старт» и «Стоп». При нажатии на кнопку «Старт» включается Timer1 и квадрат начинает заполняться случайными точками красного цвета, а точки, попавшие внутрь круга, рисуем **синими**. При нажатии на кнопку «Стоп» Timer1 выключается.



II. Написание кода программы.



1. В данном приложении будут использоваться переменные, обозначающие общее количество точек n , число точек, попавших внутрь круга m , число P_i , следовательно, их следует описать в общем разделе объявления переменных.

```
Dim m, n, pi
```

2. В событийной процедуре загрузки формы необходимо включить генератор случайных чисел и изменить масштаб Окна рисунка с помощью метода `Scale`, указав реальные координаты левого нижнего угла и правого верхнего.

```
Private Sub Form_Load()  
Randomize  
pic.Scale (-1, -1) - (1, 1)  
End Sub
```

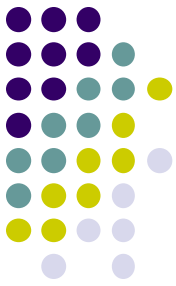
3. В событийной процедуре Щелчок на Кнопку «Старт» необходимо запустить `Timer1`, изменив его свойство `Enabled` на `True`.

В окне рисунка, используя методы `Line` и `Circle`, нарисовать квадрат со стороной 2 и окружность единичного радиуса, вписанную в данный квадрат.

```
Private Sub CommandButton1_Click()  
timer1.Enabled = True  
pic.Line (-1, -1) - (1, 1), , B  
pic.Circle (0, 0), 1  
End Sub
```

← Признак рисования прямоугольника (Bar) методом `Line`

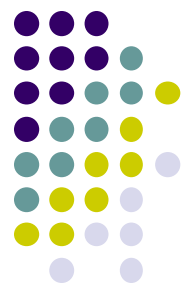
4. Каждую миллисекунду срабатывает событие Timer1, в результате которого должна быть «брошена» в квадрат случайная точка **красного цвета** с координатами x, y . Таким образом, счетчик точек (n) увеличивается на 1, формируются случайные (внутри квадрата) координаты точки и проверяется принадлежность этой точки вписанному в квадрат кругу. Если точка попадает в круг, то возрастает на 1 другой счетчик (m). Точки внутри круга для наглядности изображаются другим (**синим**) цветом.



```
Private Sub timer1_Timer()  
    n = n + 1 ' общее число точек, попавших в квадрат  
    x = Rnd * 2 - 1 ' получение случайной координаты x в интервале [ -1,+1 ]  
    y = Rnd * 2 - 1 ' получение случайной координаты y в интервале [ -1,+1 ]  
    If (x * x + y * y) <= 1 Then ' проверка условия принадлежности точки кругу  
        m = m + 1 ' число точек, попавших в круг  
        pic.PSet (x, y), RGB(0, 0, 255) ' рисование синей точки внутри круга  
    Else  
        pic.PSet (x, y), RGB(255, 0, 0) ' рисование красной точки вне круга  
    End If  
    pi = m * 4 / n ' вычисление и вывод числа ПИ в каждый момент  
    Label1.Caption = "m=" + Str(m) ' вывод числа m в каждый момент  
    Label2.Caption = "n=" + Str(n) ' вывод числа n в каждый момент  
    label3.Caption = pi  
End Sub
```

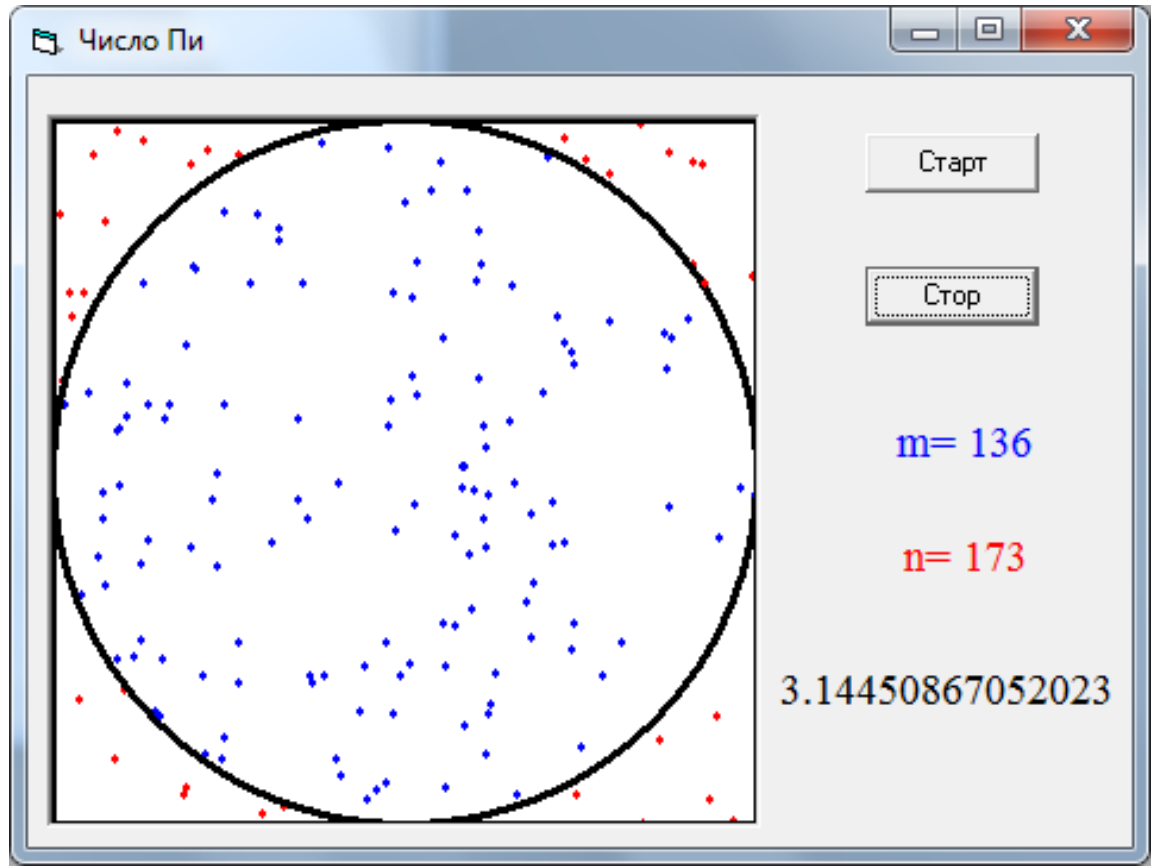
Str - функция преобразования числа в строку

5. В событийной процедуре Щелчок на Кнопку «Стоп» необходимо выключить Timer1 и вывести приближенное значение числа ПИ.



```
Private Sub CommandButton2_Click()  
    timer1.Enabled = False  
    label3.Caption = pi  
End Sub
```

Результат статистического моделирования вычисления числа π в течении нескольких секунд работы программы представлен на рисунке.



Ресурсы:

При разработки этого примера была использована статья
Паньгиной Нины Николаевны и Паньгина Андрея Александровича –
«Статистическое моделирование: метод Монте-Карло.»
Школа современного программирования 2002г.

