

7 класс. Практическая работа №2 (Turtle)

Тема: Цикл for



Ход работы

1. Выполняя предыдущую практическую работу вы не могли не заметить, что при составлении программ рисования n-угольников приходилось многократно вводить одни и те же повторяющиеся команды. Для организации повторяющихся действий в языке python имеется специальная инструкция. Которая называется цикл. Разновидностей циклов в python два. Сегодня мы рассмотрим один из них — это **цикл for**. Правила записи различных конструкций языка называется *синтаксисом*. Синтаксис цикла for имеет следующий вид:

```
for j in range(n):
```

Здесь *j* — переменная цикла, а *range(n)* — функция генерирующая арифметическую прогрессию. Если эта функция имеет один аргумент (как в нашем случае), то количество членов прогрессии будет = *n* (а знаменатель прогрессии будет = 1). На каждом шаге цикла переменная *j* будет принимать очередное (на единицу большее) число этой прогрессии. Всего будет выполнено *n* — шагов от 0 до *n*-1. Убедимся в этом, выполнив небольшой код:

```
# Программа 2.1
from turtle import*
reset()
up()
for j in range(10):
    write(j,
          True,
          font=("Arial",20, "bold"))
mainloop()
```

0123456789

Функция `write()` печатает на холсте некоторый текст. Так как аргументом этой функции выступает переменная цикла, то Черепаха напишет на холсте числовую последовательность от 0 до 9.

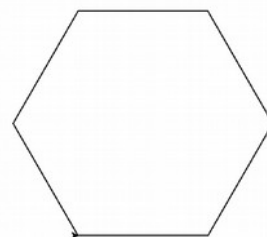
Примечание: Строки заключенные в (), [] и {} скобки можно разрывать и переносить на новую строку, это не будет являться ошибкой.

Функция `mainloop()` не позволит окну закрыться автоматически после

выполнения программы (понадобится тем разработчикам, которые используют в своей работе среду Eclipse).

2. Сделаем программу рисования шестиугольника более лаконичной:

```
# Программа 2.2
from turtle import*
reset()
speed(10)
for j in range(6):
    fd(200)
    lt(60)
mainloop()
```

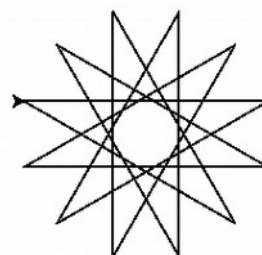


Обратите внимание на отступы в теле цикла. **Одна или более инструкций, входящие в тело цикла, должна иметь отступы слева, одинаковой величины (условимся, что отступ будет равен однократному нажатию на клавишу Tab).**

Примечание. Обычно, при переходе на новую строку, среда (IDLE или Eclipse) делает такой отступ автоматически

3. Нарисуем с помощью цикла звездообразный многоугольник. В следующей программе Черепаха нарисует на холсте 12-ти конечную звезду:

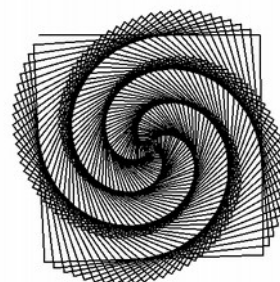
```
# Программа 2.3
from turtle import*
reset()
speed(10)
for j in range(12):
    fd(200)
    rt(150)
mainloop()
```



Для рисования звезд с нечетным количеством лучей воспользуйтесь правилом для угла поворота: $\alpha = 180 - 360 / (2 * n)$.

4. Изменение переменной цикла во время его работы можно использовать с пользой. В программе ниже мы воспользуемся этим для создания вот такой стильной спирали:

```
# Программа 2.4
from turtle import*
reset()
speed(10)
pensize(2)
for j in range(300):
    fd(300-j)
```



```
rt(89)
mainloop()
```

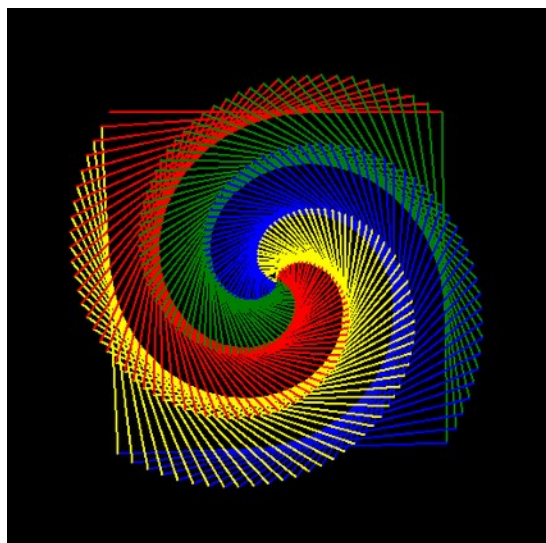
Идея алгоритма очень проста: вместо того, чтобы рисовать равные стороны квадрата мы на каждом шаге алгоритма уменьшаем сторону на 1 пиксель (с помощью переменной цикла *j*) и выполняем поворот не на 90°, а на 1° меньше, т. е. на 89°.

5. Усложним задачу: раскрасим каждую спираль в определенный цвет. Но вот незадача! Во время работы цикла мы не можем его остановить, для того, чтобы задать новый цвет. Немножко схитрим. Введем в нашу программу список цветов `Colors`. `Colors` — это массив, а в массиве каждый элемент имеет свой индекс, т. е. позицию; индексация начинается с нуля. Например:

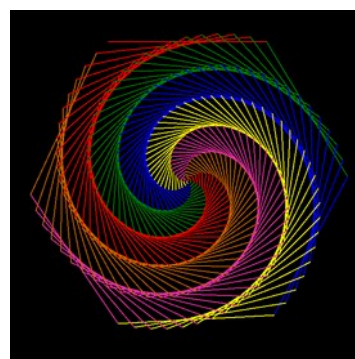
```
Colors = ["red", "green", "blue", "yellow"]
```

`Colors[0]` — имеет значение `red`, `Colors[1]` - `green` и т. д. Поскольку всего спиралей - 4, то производя целочисленное деление (с помощью операции `%`) переменной цикла (*j*) на 4 мы будем получать остаток от деления, причем на каждом шаге новый — от 0 до 3, а это и будут индексы (позиции) цветов в массиве. На черном фоне цветные спирали кажутся весьма изящными:

```
# Программа 2.5
from turtle import*
reset()
bgcolor("black")
speed(10)
pensize(2)
Colors = ["red", "green",
          "blue", "yellow"]
for j in range(300):
    pencolor(Colors[j%4])
    fd(300-j)
    rt(89)
mainloop()
```



6. Составьте программу для черепахи, которая будет рисовать на холсте цветные спирали числом 3, 5 и 6. Пример для шести спиралей:



Домашняя работа

Составьте программы рисования следующих рисунков.

