Алгоритмы. Решение автоподсчетом точек и математических формул.

Учебный исполнитель Черепаха

Исполнитель **Черепаха** перемещается на экране компьютера, оставляя след в виде линии, если предварительно была выполнена команда поднять хвост. Для передвижения без рисования линии необходимо выполнить команду опустить хвост. В каждый конкретный момент известно положение исполнителя и направление его движения.

Команды исполнителя Черепаха

penup — включает режим бесследного перемещения исполнителя.

pen — включает режим, позволяющий оставлять Черепахе след в виде линии.

forward (n) — вызывает передвижение Черепахи на n шагов в направлении движения.

backward (k) — вызывает передвижение Черепахи на k шагов против направления движения.

left (a) — изменяет направление движения на а градусов против часовой стрелки.

right (b) — изменяет направление движения на b градусов по часовой стрелке.

Циклы в среде программирования КуМир

Общий вид цикла N раз:

for i in range(N): тело_цикла

Ключевое слово for обозначает начало циклического алгоритма. Параметр N, являющийся целым выражением, задает число повторений команд, составляющих тело цикла.

Пример задания

Исполнитель Черепаха действует на плоскости с декартовой системой координат. В начальный момент Черепаха находится в начале координат, ее голова направлена вдоль положительного направления оси ординат, хвост опущен. При опущенном хвосте Черепаха оставляет на поле след в виде линии. В каждый конкретный момент известно положение исполнителя и направление его движения. У исполнителя существует две команды: Вперед п (где n – целое число), вызывающая передвижение Черепахи на n единиц в том направлении, куда указывает ее голова; Направо m (где m – целое число), вызывающая изменение направления движения на m градусов по часовой стрелке.

Запись Повтори k [Команда1 Команда2 ... КомандаS] означает, что последовательность из S команд повторится k раз.

Черепахе был дан для исполнения следующий алгоритм:

Повтори 123 [Вперед 111 Направо 120].

Определите, сколько точек с целочисленными координатами будут находится внутри области, ограниченной линией, заданной данным алгоритмом. Точки на линии учитывать не следует.

Решение

- 1 Импортируем библиотеку turtle, с помощью которой будем рисовать. Так как по умолчанию в библиотеке исполнитель смотрит вправо, поворачиваем его влево на 90 градусов.
- **2** Фигурой, которую рисует алгоритм, является треугольник, значит, необходимо сделать три итерации.
- 3 Считываем холст, чтобы посчитать количество точек на нем, функция canvas.find_overlapping(x, y, x, y) находит пересечения прямоугольника с координатами x0, y0, x1, y1 с фигурами на холсте. Если передать координаты x, y, x, y получится точка, тогда мы будем искать, пересекает ли точка какую-либо фигуру.
- 4 Если в tmp записалось только 5, тогда точка пересекает только заливку, следовательно, обновляем значение переменной.

Полный код:

```
import turtle
t = turtle.Pen()
t.speed(10000)
k = 100
t.left(90)
t.begin_fill()
for i in range(3):
    t.forward(111 * k)
    t.right(120)
    t.end_fill()
canvas = turtle.getcanvas()
for x in range(-400 * k, 400 * k, k):
    for y in range(-400 * k, 400 * k, k):
        tmp = canvas.find_overlapping(x, y, x, y)
        if tmp == (5, ):
            cnt += 1
turtle.mainloop()
```

Заметки	