

Исполнитель Цапля действует на плоскости с декартовой системой координат. В начальный момент Цапля находится в начале координат, её клюв направлен вдоль положительного направления оси ординат, клюв опущен. При опущенном клюве Цапля оставляет на поле след в виде линии. В каждый конкретный момент известно положение исполнителя и направление его движения. У исполнителя существует три команды: Вперёд n (где n — целое число), вызывающая передвижение Цапли на n единиц в том направлении, куда указывает её клюв; Направо m (где m — целое число), вызывающая изменение направления движения на m градусов по часовой стрелке; Дуга r, a, b, α (где r, a, b, α — целые числа), вызывающая передвижение Цапли из текущей точки с

координатами (x, y) по дуге окружности с центром в точке с координатами (a, b) и радиусом r , градусная мера дуги равна α , движение по дуге идёт по часовой стрелке.

Запись **Повтори k [Команда 1 Команда 2 ... Команда S]** означает, что последовательность из S команд повторится k раз.

Цапле был дан для исполнения следующий алгоритм:

Направо 180 Вперёд 5 Направо 90 Вперёд 50 Направо 90 Вперёд 5 Повтори 5 [Дуга 5, 5, 0, 180].

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

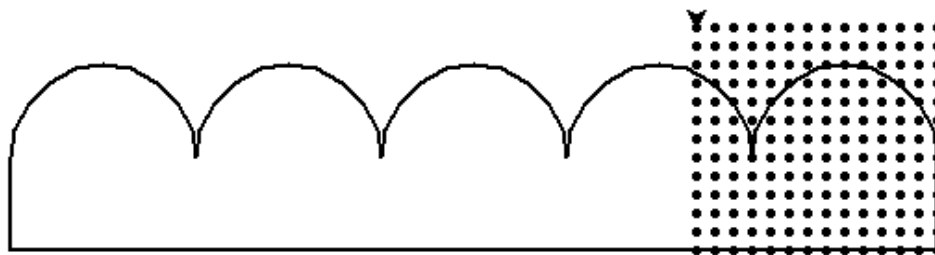
Решение.

Приведём решение на языке Python.

```
import turtle
t=turtle.Turtle() #или import turtle as t вместо этой строки

t.reset()
t.seth(90)
t.width(2)
t.speed(20)
k = 10 #коэффициент для увеличения масштаба
t.right(180)
t.forward(5*k)
t.right(90)
t.forward(50*k)
t.right(90)
t.forward(5*k)
for i in range(5):
    t.seth(90)
    t.circle(-5*k,180)
t.penup()
for x in range(0,-14,-1): #точки
    for y in range(-5,8):
        t.goto(x*k , y*k )
```

```
t.dot(5)  
turtle.mainloop()
```



Посчитаем количество точек в одном секторе (75) и умножим их на количество секторов (5), плюс количество точек на границе секторов (по четыре на границу). Итоговый ответ $75 \cdot 5 + 4 \cdot 4 = 391$.

Ответ: 391.