

# Черепашка. Позиционирование

Цикл while

#### Позиционирование черепашки

Повторение

Положение черепашки в окне

Изменение координат

Текущие координаты черепашки

# Повторение 🕵

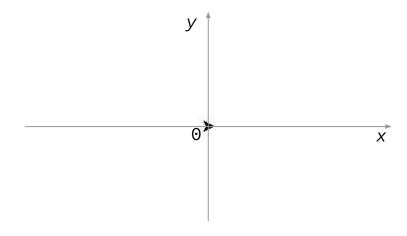
Если вы забыли, как работать с черепашкой, то можете изучить эти материалы:

Создание и управление черепашкой, поднять/опустить перо, очищение холста

Скорость, контуры, цвет пера и заливка фигур

### Положение черепашки в окне

Место, где появляется черепашка в начале, имеет координаты (0, 0). Координата "х" увеличивается вправо и уменьшается влево. Координата "у" увеличивается вверх и уменьшается вниз.



# Изменение координат

Для перемещения черепашки относительно координат есть несколько методов.

goto(x, y) — переместить черепашку в точку с координатами (x, y). Обратите внимание, что она продолжит оставлять след, если не поднять перо.



setx(n) — задать координате х значение n.

sety(n) — задать координате у значение n.

home() — переместить черепашку в точку (0,0).



## Текущие координаты черепашки

*xcor()* — узнать текущую координату х черепашки



*ycor()* — узнать текущую координату у черепашки

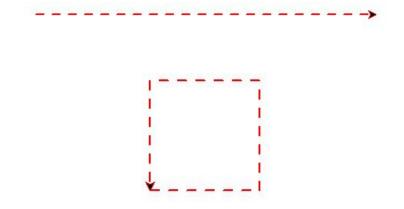


pos() — вернет сразу два числа — координату х и у.

#### Пунктирные линии 🥉



- Переместите черепашку в точку (-200, -200) и нарисуйте пунктирную линию, пока координата х <= 200.
- 2. По аналогии с предыдущим заданием нарисуйте квадрат из пунктирных линий. Используйте цикл while 4 раза.

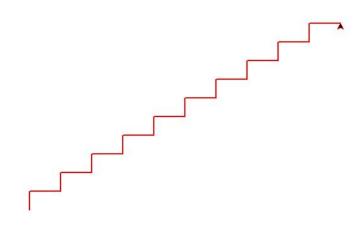


# 3. Ступеньки 🥉

Нарисуйте ступеньки.

В начале поставьте черепашку в точку слева внизу экрана.

Потом, пока одна из координат черепашки не станет больше 200, рисуйте ступеньку.



#### 4. Забор 💆

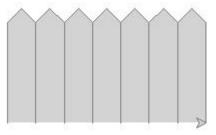


Нарисуйте забор.

В начале спросите, сколько "заборинок" нужно нарисовать, и сохраните это число в переменной n.

Затем создайте еще одну переменную і, которая будет определять номер "заборинки". В начале i = 1.

Пока і меньше или равно п, нужно рисовать "заборинку" и увеличивать і на 1.



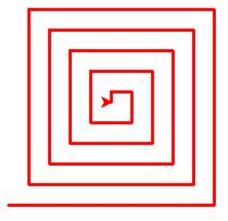
#### 5. Лабиринт 💆

Нарисуйте вот такой лабиринт.

На первый взгляд это может показаться сложным, но на самом деле алгоритм очень простой:

Создайте переменную n и запишите туда число 200 (можно взять другое или вообще спрашивать с помощью int(input())).

Пока n > 0, нужно идти вперед на n пикселей, потом поворачиваться влево, а n уменьшать, например, на 10.



# 6. Круги 🥇

Разноцветные круги.

По аналогии с предыдущим заданием нарисуйте несколько кругов разного радиуса. Для того, чтобы сделать цвет фона и цвет линии случайным, импортируйте модуль random и используй команды

t.fillcolor(random.random(), random.random(), random.random())

t.pencolor(random.random(), random.random(), random.random())

