



#ШПАРГАЛОЧКИ



# ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ PYTHON

## Базовый уровень

Материалы подготовлены отделом методической  
разработки

Больше полезных материалов и общения в нашем комьюнити  
в Telegram: [https://t.me/hw\\_school](https://t.me/hw_school)





# Создание игры Танки (Часть 1)





Чтобы запрограммировать реалистичное управление спрайтом по клавишам, необходимо выполнить два действия:

1. Создать логические переменные для хранения состояния клавиш (нажата или нет)
2. По нажатию на клавишу менять значение соответствующей ей переменной на **True**, а отвечающей за движение в противоположную сторону - на **False**. А также менять значение переменных, отвечающих за скорость движения/угол поворота.



На примере поворота влево:

```
class Game(arcade.Window):
```

```
...
```

```
def on_key_press(self, key, modifiers):
```

```
    if key == arcade.key.LEFT :
```

```
        self.green.change_angle = 2.5
```

```
        self.left_pressed = True
```

```
        self.right_pressed = False
```



При этом когда клавиша отпущена, скорость движения/угол поворота должны обнуляться, только если клавиша, отвечающая за движение в противоположную сторону, не нажата

...

```
def on_key_release(self, key, modifiers):
```

```
    if key == arcade.key.LEFT and not self.right_pressed:
```

```
        self.green.change_angle = 0
```



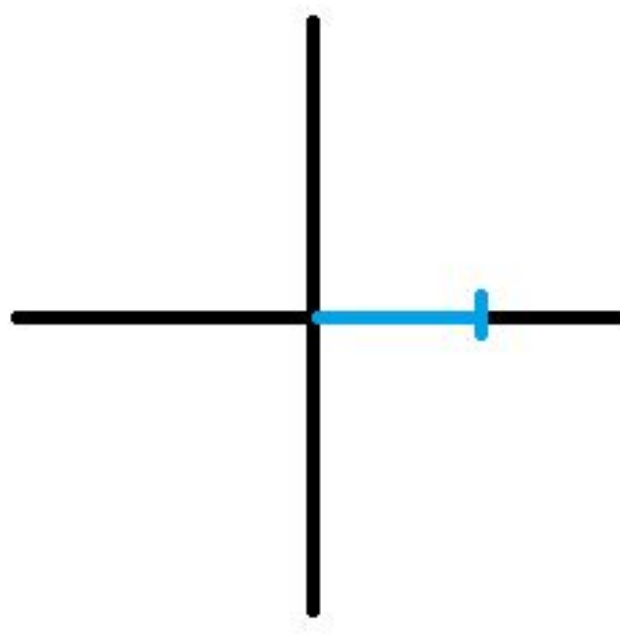
Чтобы спрайт мог не только поворачиваться, но и двигаться в нужном направлении (а не только влево-вправо и вверх-вниз), нам нужно подобрать правильное соотношение скорости по **X** и **Y**.



Для этого потребуется подключить библиотеку **math** - в ней есть нужные нам функции для расчета **синуса** и **косинуса**.

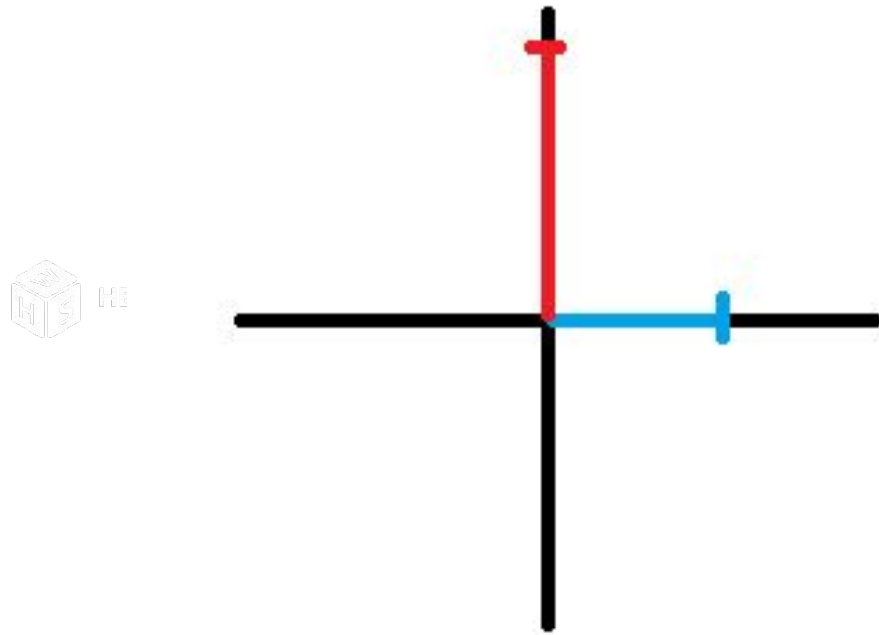


**Косинус** считает, как сильно и в какую сторону (влево или вправо) должно тянуть спрайт по **горизонтали**:





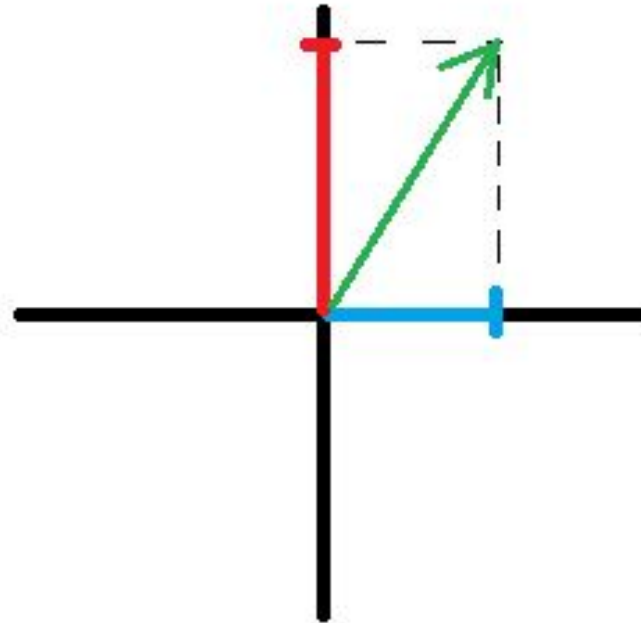
**Синус** считает, как сильно и в какую сторону (вверх или вниз)  
должно тянуть спрайт по **вертикали**:







А если найти пересечение этих двух “тяг”, то это и будет  
нужно направление





Синус и косинус считаются помощью функций **math.sin** и **math.cos**, которые принимают угол поворота в **радианах**.

**Радиан** - это другая, отличная от градусов, мера измерения угла. И чтобы превратить градус в радиан, нужна еще одна функция из математической библиотеки - **math.radians**:

# в update спрайта

```
self.part_x = math.cos(math.radians(self.angle)) # тяга по X
```

```
self.part_y = math.sin(math.radians(self.angle)) # тяга по Y
```



И теперь на эти “тяги” нужно умножить скорость движения по **X** по **Y**:

```
self.center_x += self.change_x * self.part_x
```

```
self.center_y += self.change_y * self.part_y
```



Если нам нужно сделать снаряд, движущийся в том же направлении, что и выпустивший его спрайт, снова пригодятся тяги. А именно - тяги соответствующего спрайта, их нужно умножить на скорость снаряда.



Кстати, чтобы использовать объект из другого модуля в классе нашего снаряда, нужный объект нужно передать как аргумент конструктора `__init__`



# в классе снаряда

```
def __init__(self, image, tank):
```

```
    self.angle = tank.angle # поворачиваем снаряд в том  
    же направлении, что и танк
```

```
    self.change_x = 12 * tank.part_x
```

```
    self.change_y = 12 * tank.part_y
```