## Модуль location

### Метод start\_position

Тест №1

Цель: проверка постановки персонажа на платформе при стандартных значениях

Тип: позитивный

Входные данные: x = 32, y = 416

Ожидаемый результат: 1 (по иксу на платформе), 1 (по игреку на платформе), true (не ушел за карту) и остался при таких же x, y (то есть не произошло столкновение)

Описание процесса: чтобы персонаж не проходил сквозь платформу его координаты при столкновении возвращаются на шаг назад, чтобы избежать столкновения. Данным тестом проверяем, что персонаж стоит на платформе, не выше, ни ниже

### Метод insideTexture

Тест №2

Цель: реакция на значения, при которых персонаж уже провалился под платформу и находится под текстурой

Тип: позитивный

Входные данные: x = 96, y = 480

Ожидаемый результат: 0 (по игреку не на платформе), true (не ушел за карту) и y = 416, при этом x = 96 (то есть не изменился)

Описание процесса: если персонаж провалился в текстуру, то по игреку он должен вернуться обратно

### Метод colRight

Тест №3

Цель: реакция на значения, при которых персонаж уже проходит что-то насквозь

Тип: позитивный

Входные данные: x = 1504, y = 608

Ожидаемый результат: true (не ушел за карту) и x = 1408

Описание процесса: если персонаж сквозь что-то по горизонтали, то возвращается на одну клетку назад

### Метод negative

Тест №4

Цель: правильная реакция программы на нестандартную позицию персонажа

Тип: негативный

Входные данные: x = -33, y = -60

Ожидаемый результат: 0 (за границами карты по иксу), 0(за границами карты по игреку), false (ушел за карту и персонаж умирает)

Описание процесса: если персонаж уходит за границы карты, то он автоматически умирает и игра заканчивается

## Модуль *arrow\_of\_archer*

### Метод *onDistance*

1. Тест №5
2. Цель: добиться того, чтобы лучники не стреляли, пока персонаж не подойдет на достаточную дистанцию
3. Тип: позитивный
4. Входные данные: x = 516, y = 96
5. Ожидаемый результат: 0 (лучник не выстрелил ни одной стрелы), true (персонаж живой и не вышел за границы)
6. Описание процесса: лучники начинают стрелять в персонажа, когда тот подходит на расстоянии меньше, чем 380 в данном тесте расстояние немного больше

### Метод *shots*

1. Тест №6
2. Цель: правильная стрельба лучников
3. Тип: позитивный
4. Входные данные: x = 608, y = 256, time = 3000
5. Ожидаемый результат: 12 (за указанное время лучник должен выстрелить ровно 12 раз), true (персонаж живой и не вышел за границы)
6. Описание процесса: лучники начинают стрелять в персонажа, когда тот подходит на расстоянии меньше, чем 380 в данном тесте расстояние меньше и стреляют с определенной задержкой

### Метод *negative*

1. Тест №7
2. Цель: правильная реакция на нестандартные значения времени, чтобы лучники не стреляли
3. Тип: негативный
4. Входные данные: x = 608, y = 256, time = -3000
5. Ожидаемый результат: 0 (лучник не должен стрелять), true (персонаж живой и не вышел за границы)
6. Описание процесса: лучники начинают стрелять в персонажа, когда тот подходит на расстоянии меньше, чем 380 в данном тесте расстояние меньше и стреляют с определенной задержкой

## Модуль update

### Метод *jump*

Тест №8

Цель: добиться правильного передвижения при прыжке персонажа

Тип: позитивный

Входные данные: x = 32, y = 416, time = 3000

Ожидаемый результат: x = 32 (не произошло движения при прыжках в течении данного времени)

Описание процесса: при прыжке придается скорость персонажу и необходимо проверить, что координаты меняются только по игреку, если скорость по иксу нулевая

### Метод *jump*

Тест №8

Цель: добиться правильного передвижения при прыжке персонажа

Тип: позитивный

Входные данные: x = 32, y = 416, time = 3000, dy = -0.3

Ожидаемый результат: x = 32 (не произошло движения при прыжках в течении данного времени)

Описание процесса: при прыжке придается скорость персонажу и необходимо проверить, что координаты меняются только по игреку, если скорость по иксу нулевая

### Метод *negative*

Тест №9

Цель: добиться правильного передвижения при прыжке персонажа(нестандартное значение)

Тип: негативный

Входные данные: x = 32, y = 416, time = 3000, dy = 0,3

Ожидаемый результат: x = 32 (не произошло движения при прыжках в течении данного времени), alive = false (персонаж ушел за границы)

Описание процесса: персонаж находится уже на платформе и ему придается положительная скорость по игреку, то есть он прыгает вниз, в текстуру