**Тестирование метода checkTriangle()**

1. Проверка корректной работы метода при значении размера стороны, равном 0, или отрицательному числу:

|  |  |
| --- | --- |
| Условия | Ожидаемый результат |
| **a = 0, или a < 0; b > 0: c > 0.** | false, triangle.getMessage().equals(“a<=0”); |
| **b = 0, или b < 0; a > 0: c > 0.** | false, triangle.getMessage().equals(“b<=0”); |
| **c = 0, или c < 0; a > 0: b > 0.** | false, triangle.getMessage().equals(“c<=0”); |

1. Проверка корректной работы метода при значении размеров сторон, при которых не соблюдается условие существования треугольника

|  |  |
| --- | --- |
| Условия | Ожидаемый результат |
| **a + b = c или a + b < c;** | false, triangle.getMessage().equals(“a+b<=c”); |
| **a + c = b или a + c < b;** | false, triangle.getMessage().equals(“a+c<=b”); |
| **b + c = a или b + c < a;** | false, triangle.getMessage().equals(“b+c<=a”); |

1. Поскольку в методе есть сложение, необходима проверка на переполнение

|  |  |
| --- | --- |
| Условия | Ожидаемый результат |
| **a = b = c = Double.MAX\_VALUE** | Exception |

1. Проверка позитивных сценариев

|  |  |
| --- | --- |
| Условия | Ожидаемый результат |
| **a + b > c; a,b,c >=0** | true, triangle.getMessage().equals(“”); |

**Тестирование метода detectTriangle()**

1. Проверка корректной работы метода при значении размера стороны, равном 0, или отрицательному числу.

Должно генерироваться исключение (Разработчик должен предусмотреть невозможность вызова метода detectTriangle() до вызова метода checkTriangle()):

|  |  |
| --- | --- |
| Условия | Ожидаемый результат |
| **a = 0, или a < 0; b > 0: c > 0.** | Exception |
| **b = 0, или b < 0; a > 0: c > 0.** | Exception |
| **c = 0, или c < 0; a > 0: b > 0.** | Exception |

1. Проверка корректной работы метода при значении размеров сторон, при которых не соблюдается условие существования треугольника.

Должно генерироваться исключение (Разработчик должен предусмотреть невозможность вызова метода detectTriangle() до вызова метода checkTriangle()):

|  |  |
| --- | --- |
| Условия | Ожидаемый результат |
| **a + b = c или a + b < c;** | Exception |
| **a + c = b или a + c < b;** | Exception |
| **b + c = a или b + c < a;** | Exception |

1. Поскольку в методе есть умножение, необходима проверка на переполнение

|  |  |
| --- | --- |
| Условия | Ожидаемый результат |
| **a = b = c = Double.MAX\_VALUE** | Exception |

1. Проверка сценариев комбинаций сторон различных видов треугольников

|  |  |
| --- | --- |
| Условия | Ожидаемый результат |
| **Прямоугольный** | 8 |
| **Прямоугольный равнобедренный** | 10 |
| **Равносторонний [равнобедренный]** | 3 |
| **Равнобедренный** | 2 |
| **Обычный** | 4 |

**Тестирование метода getSquare()**

1. Проверка корректной работы метода при значении размера стороны, равном 0, или отрицательному числу.

Должно генерироваться исключение (Разработчик должен предусмотреть невозможность вызова метода getSquare() до вызова метода checkTriangle()):

|  |  |
| --- | --- |
| Условия | Ожидаемый результат |
| **a = 0, или a < 0; b > 0: c > 0.** | Exception |
| **b = 0, или b < 0; a > 0: c > 0.** | Exception |
| **c = 0, или c < 0; a > 0: b > 0.** | Exception |

1. Проверка корректной работы метода при значении размеров сторон, при которых не соблюдается условие существования треугольника.

Должно генерироваться исключение (Разработчик должен предусмотреть невозможность вызова метода detectTriangle() до вызова метода checkTriangle()):

|  |  |
| --- | --- |
| Условия | Ожидаемый результат |
| **a + b = c или a + b < c;** | Exception |
| **a + c = b или a + c < b;** | Exception |
| **b + c = a или b + c < a;** | Exception |

1. Поскольку в методе есть умножение, необходима проверка на переполнение

|  |  |
| --- | --- |
| Условия | Ожидаемый результат |
| **a = b = c = Double.MAX\_VALUE** | Exception |

1. Проверка расчета площади на валидных комбинациях сторон различных видов треугольников