

# Университет ИТМО

Факультет программной инженерии и компьютерной техники  
Направление подготовки 09.03.04 Информатика и вычислительная техника  
Дисциплина «Тестирование программного обеспечения»

## Отчет По лабораторной работе №2 Вариант 5263

Выполнили:  
*Казаченко Р. О.*  
*Степанов М. А.*  
Преподаватель:  
*Гаврилов А. В.*

Санкт-Петербург, 2023 г.

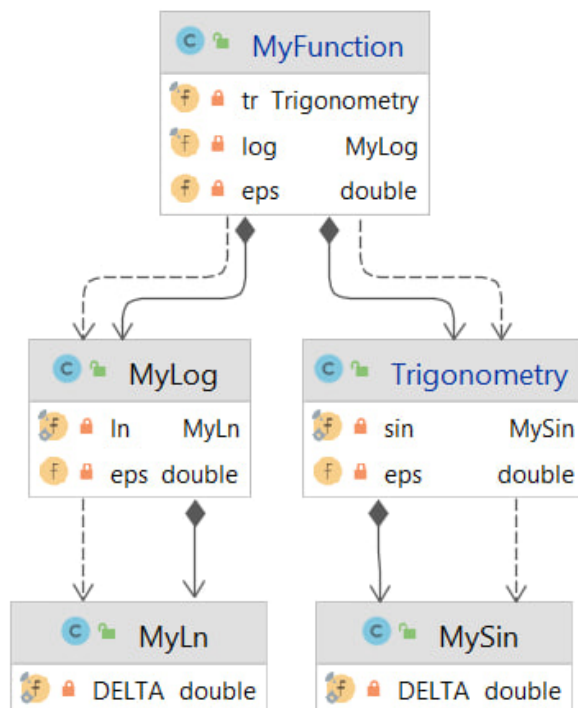
### Задание:

1. Разработать приложение, руководствуясь приведёнными правилами.
2. С помощью JUNIT4 разработать тестовое покрытие системы функций, проведя анализ эквивалентности и учитывая особенности системы функций. Для анализа особенностей системы функций и составляющих ее частей можно использовать сайт <https://www.wolframalpha.com/>.
3. Собрать приложение, состоящее из заглушек. Провести интеграцию приложения по 1 модулю, с обоснованием стратегии интеграции, проведением интеграционных тестов и контролем тестового покрытия системы функций.

### Система функций:

$$\begin{cases} \left( \left( \left( \frac{\left( \frac{\cos(x)}{\tan(x)} \right)^3 \cdot \sin(x)}{\sec(x)} \right) \cdot \left( \frac{\cos(x) - (\tan(x) \cdot \sec(x))}{\sec(x) + (\cos(x) \cdot \cot(x))} \right) \right) \right) & \text{if } x \leq 0 \\ \left( \left( \left( ((\log_{10}(x) - \log_2(x)) \cdot \log_2(x)) + \left( \frac{\log_{10}(x)}{\ln(x)} \right) \right) + \log_2(x) \right) + \left( \frac{\frac{\log_{10}(x)}{\log_2(x)}}{\log_3(x)} \right) \right) & \text{if } x > 0 \end{cases}$$

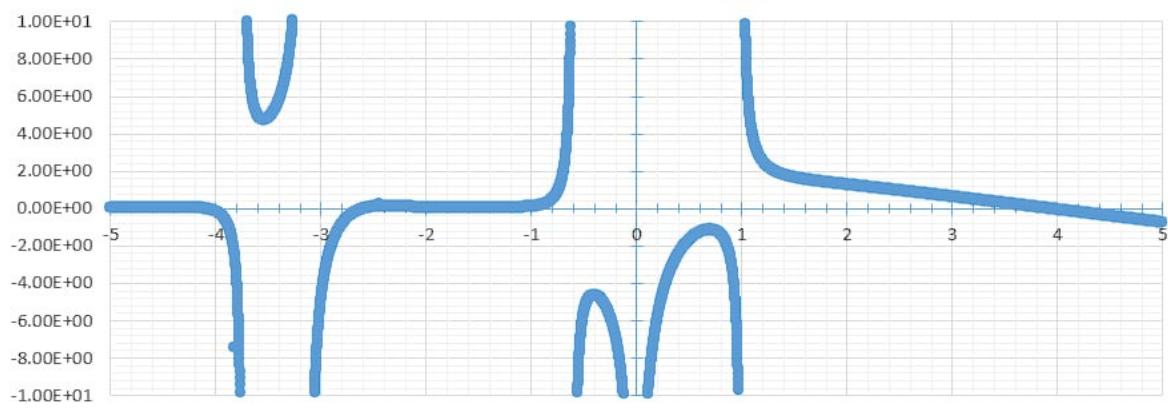
### UML-диаграмма классов разработанного приложения:



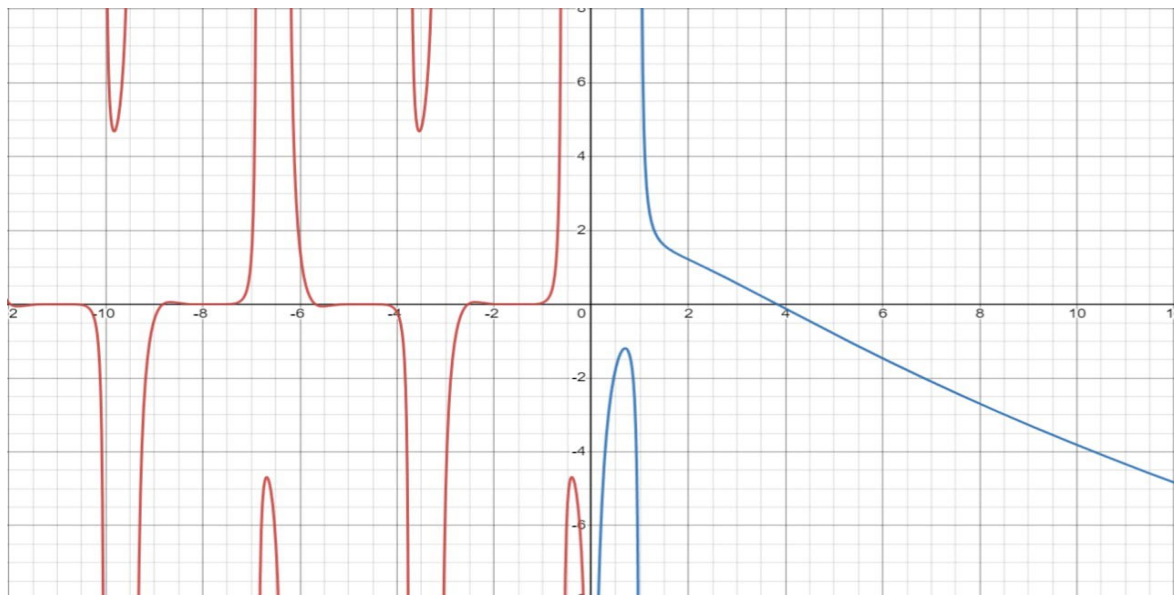
### Описание тестового покрытия с обоснованием его выбора:

Было проведено модульное тестирование всех функций в приложении, начиная с базовой ( $\sin(x)$ ), и заканчивая системой функций по варианту. Тестовое покрытие выбрано с учетом анализа эквивалентности и особенностей поведения каждой функции. Были проверены граничные и промежуточные значения, а также периодичность системы функций. Для проведения интеграционных тестов был выбран интервал значений  $[-4; 4]$  с шагом  $0.01$ , так как такой интервал является достаточным для проверки характерных областей поведения функции, в том числе мест, где она не определена. Интеграция приложения проводилась **сверху вниз**, так как зависимость модулей на верхнем уровне более комплексная, чем на нижнем.

### График функции, полученный в результате интеграции:



### Эталонный график функции:



**Исходный код:**

[https://github.com/R4MZEZ/TPO\\_lab2](https://github.com/R4MZEZ/TPO_lab2)

**Выводы:**

Во время выполнения лабораторной работы мы изучили работу классов заглушек на примере библиотеки Mockito и применили ее для интеграционного тестирования написанного нами приложения для решения системы уравнений.

