

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

ЛАБОРАТОРНАЯ РАБОТА №2

по дисциплине
“Тестирование программного обеспечения”

Студенты:

Алексеев Даниил
Сабитов Данил

Группа Р33302

Преподаватель:

Харитоновна Анастасия Евгеньевна

Санкт-Петербург
2023

Задание

Провести интеграционное тестирование программы, осуществляющей вычисление системы функций (в соответствии с вариантом).

Введите вариант:

$$\begin{cases} \left(\frac{\left(\left(\left(\sec(x)^2 + \csc(x) \right)^2 \right)^3 \right)}{\sin(x)} \right) & \text{if } x \leq 0 \\ \left(\left(\left(\left(\frac{\log_3(x) + \log_5(x)}{\frac{\ln(x)}{\log_2(x)}} \right) - \log_3(x) \right) \cdot \log_2(x) \right)^2 \right) & \text{if } x > 0 \end{cases}$$

$x \leq 0 : (((((\sec(x) ^ 2) + \csc(x)) ^ 2) ^ 3) / \sin(x))$

$x > 0 : (((((\log_3(x) + \log_5(x)) / (\ln(x) / \log_2(x))) - \log_3(x)) * \log_2(x)) ^ 2)$

Правила выполнения работы:

1. Все составляющие систему функции (как тригонометрические, так и логарифмические) должны быть выражены через базовые (тригонометрическая зависит от варианта; логарифмическая - натуральный логарифм).
2. Структура приложения, тестируемого в рамках лабораторной работы, должна выглядеть следующим образом (пример приведён для базовой тригонометрической функции $\sin(x)$):
3. Обе "базовые" функции (в примере выше - $\sin(x)$ и $\ln(x)$) должны быть реализованы при помощи разложения в ряд с задаваемой погрешностью. Использовать тригонометрические / логарифмические преобразования для упрощения функций ЗАПРЕЩЕНО.
4. Для КАЖДОГО модуля должны быть реализованы табличные заглушки. При этом, необходимо найти область допустимых значений функций, и, при необходимости, определить взаимозависимые точки в модулях.
5. Разработанное приложение должно позволять выводить значения, выдаваемое любым модулем системы, в csv файл вида «X, Результаты модуля (X)», позволяющее произвольно менять шаг наращивания X. Разделитель в файле csv можно использовать произвольный.

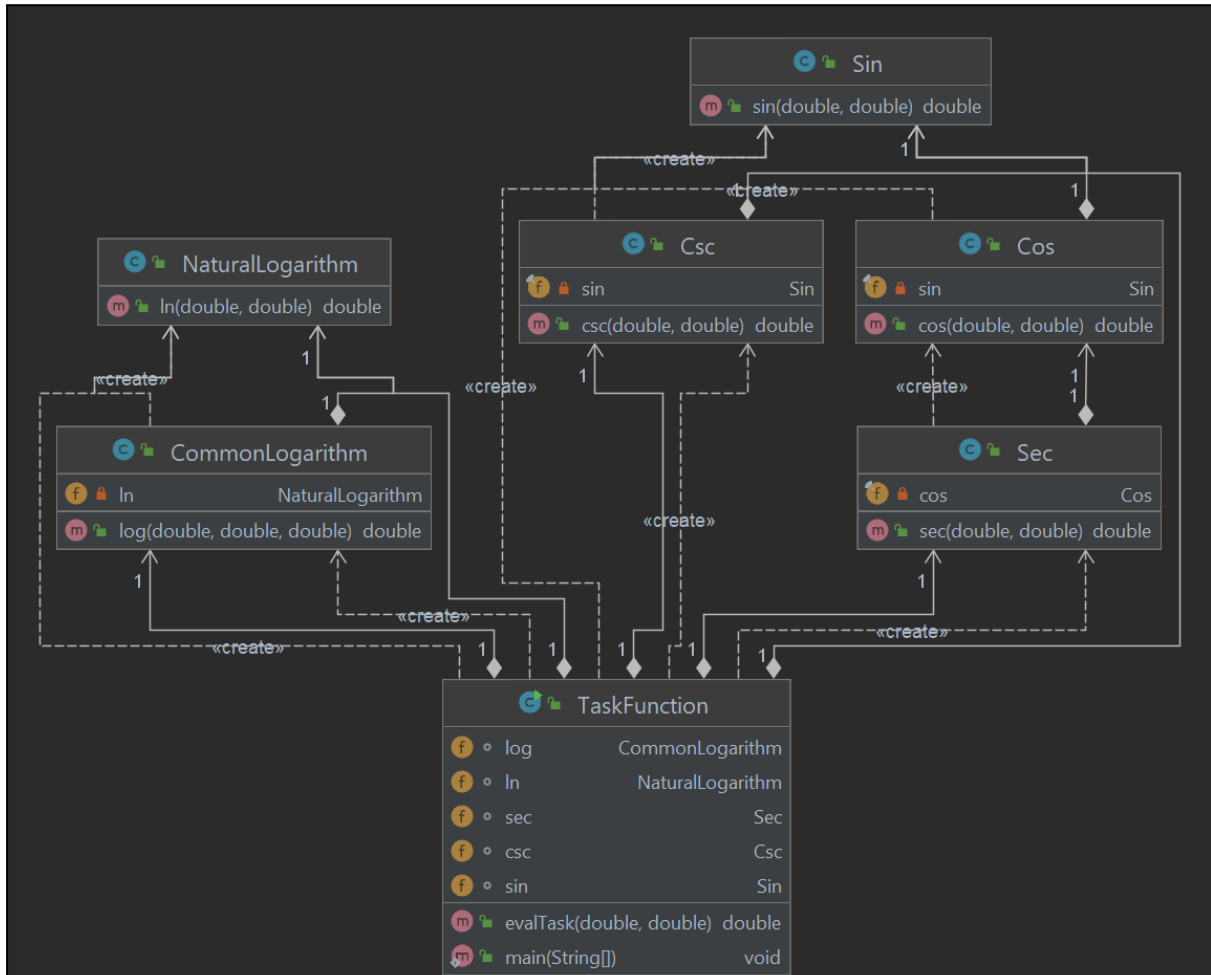
Порядок выполнения работы:

1. Разработать приложение, руководствуясь приведёнными выше правилами.
2. С помощью JUNIT4 разработать тестовое покрытие системы функций, проводя 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.

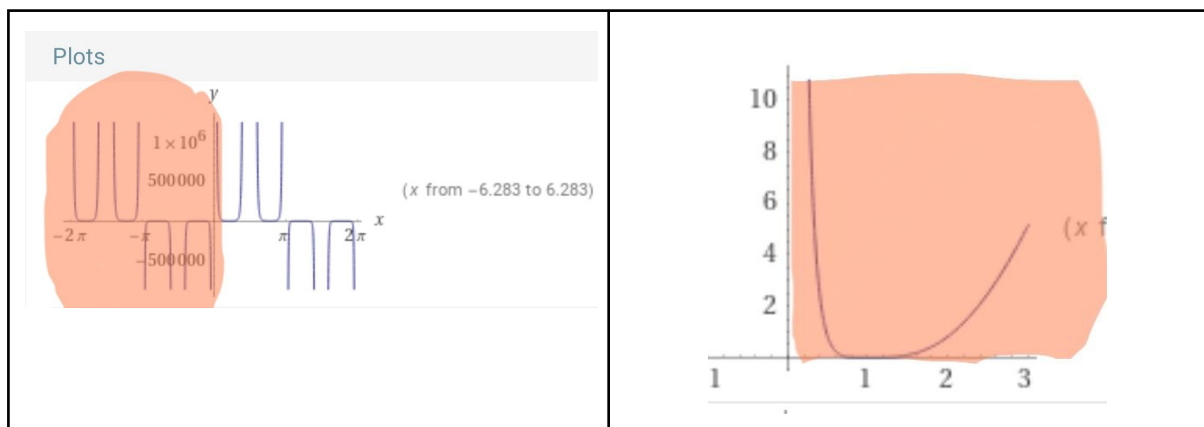
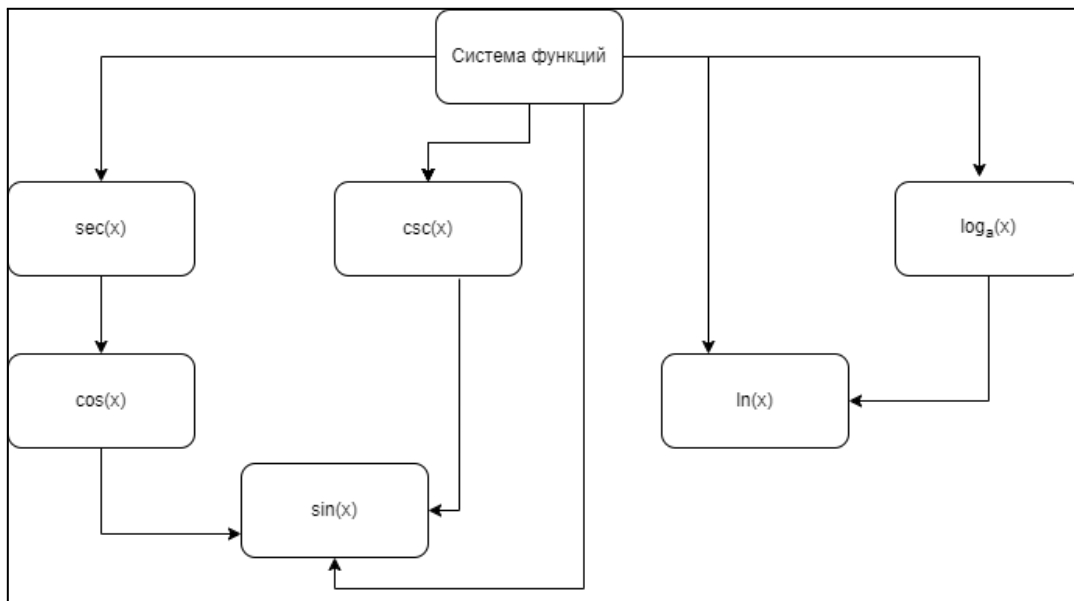
Выполнение:

ссылка на репозиторий: <https://github.com/nibitoff/Software-testing/tree/main/Lab2>

Uml - диаграмма классов:



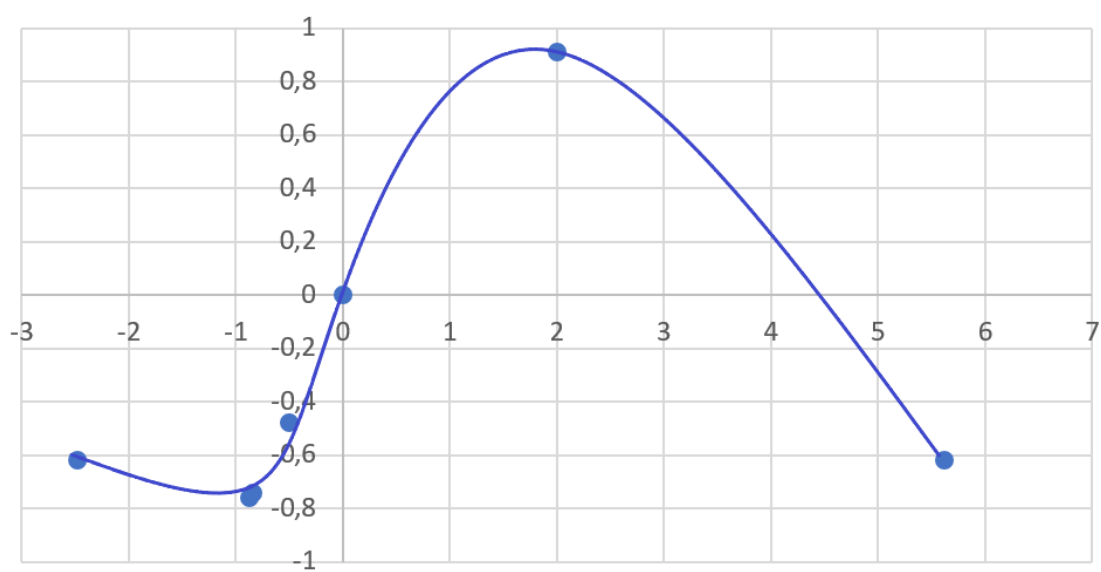
Описание тестового покрытия



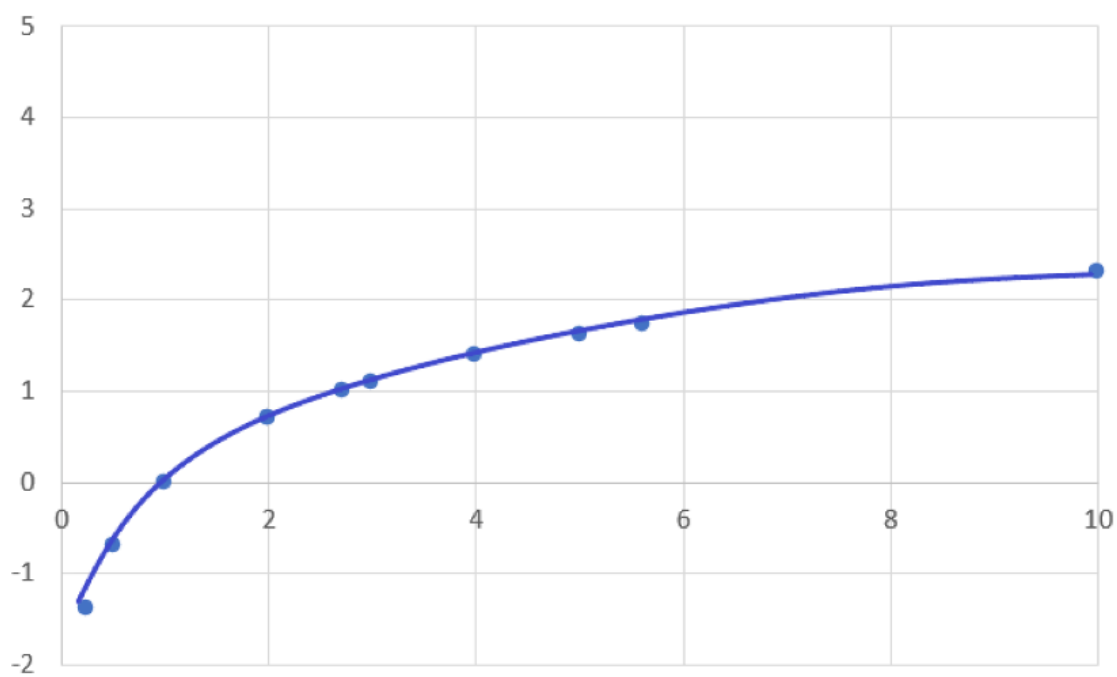
при $x \leq 0$, функция ведёт себя как периодическая, есть точки разрыва, рядом с которыми они стремятся к бесконечности. Мы брали точки рядом с точкой, где меняется знак производной. И рядом с точками разрыва функции, в которых функция стремится к бесконечности (только мы не брали на самом деле, надо попробовать добавить)

при $x > 0$, функция уже имеет только одну точку перерыва - в точке ноль. В остальных точках она определена. Мы взяли точки рядом с точкой изменения знака производной

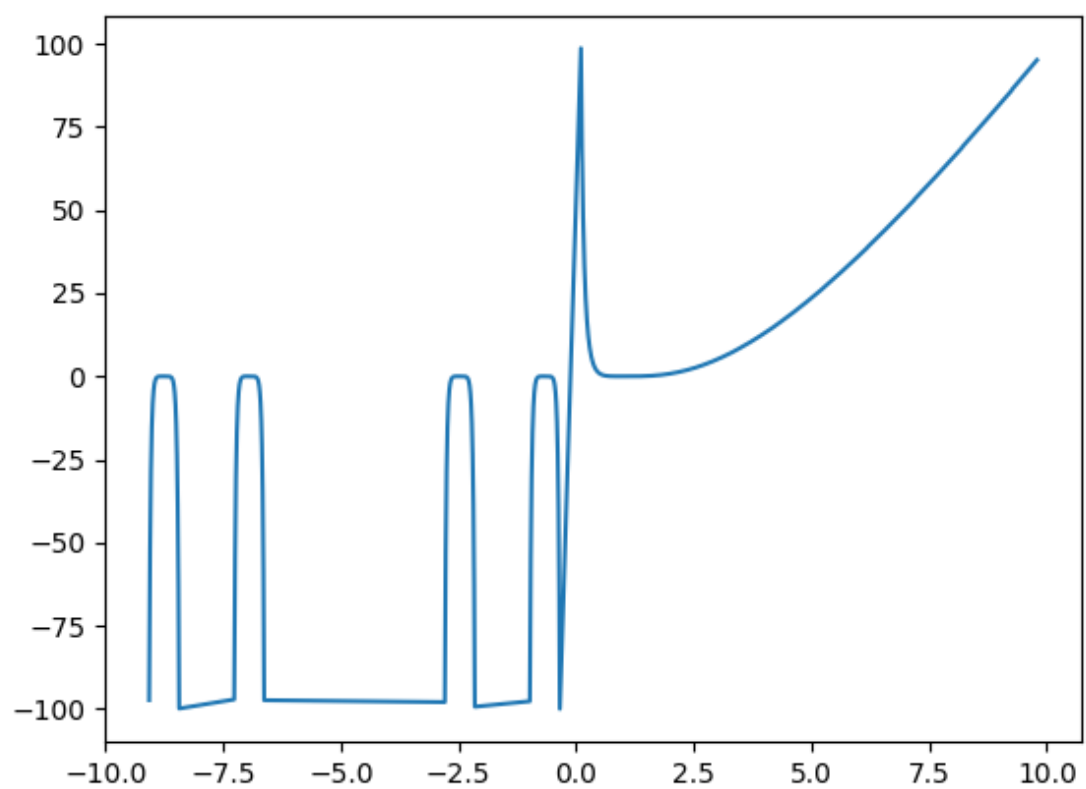
Графики, построенные csv-выгрузкам



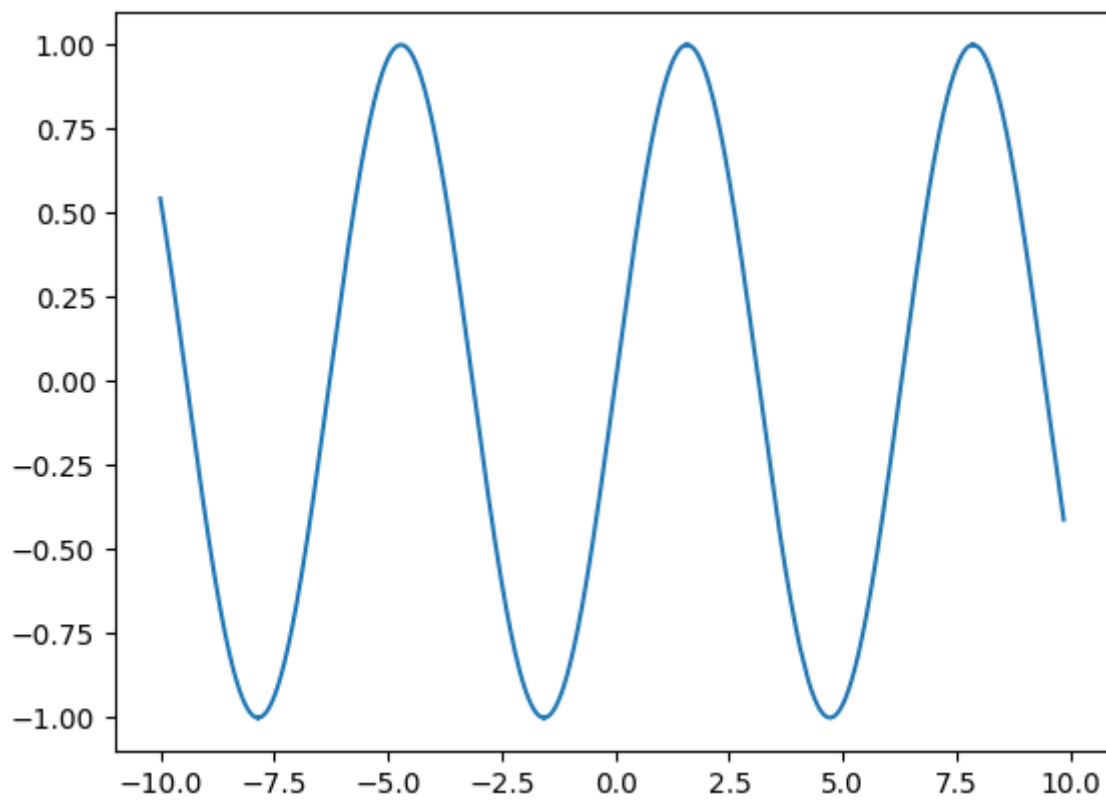
$\sin(x)$



$\ln(x)$



Только в точках разрыва график получается некрасивым



Вывод:

В ходе выполнения лабораторной работы было проведено интеграционное тестирование написанной нами программы, реализованной согласно варианту. Также мы изучили работу классов заглушек на примере библиотеки Mockito и применили ее в ходе тестирования.