УНИВЕРСИТЕТ ИТМО

Факультет программной инженерии и компьютерной техники

Направление подготовки 09.03.04 Программная инженерия

Дисциплина «Тестирование программного обеспечения»

**Лабораторная работа №2**

Вариант 13285

Студент

*Митрофанов Е. Ю.*

*Любкин А. С.*

*P33101*

Преподаватель

*Яркеев А. С.*

*Клименков С. В.*

Санкт-Петербург, 2022 г.

Задание лабораторной работы

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

Изображение выглядит как текст

Автоматически созданное описание

**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 сsv файл вида «X, Результаты модуля (X)», позволяющее произвольно менять шаг наращивания Х. Разделитель в файле csv можно использовать произвольный.

**Порядок выполнения работы:**

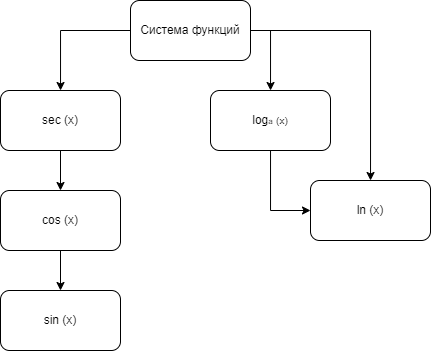
1. Разработать приложение, руководствуясь приведёнными выше правилами.
2. С помощью JUNIT4 разработать тестовое покрытие системы функций, проведя анализ эквивалентности и учитывая особенности системы функций. Для анализа особенностей системы функций и составляющих ее частей можно использовать сайт <https://www.wolframalpha.com/>.
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.

UML - диаграмма классов

Изображение выглядит как текст, внутренний, черный, снимок экрана

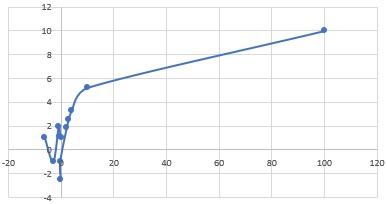
Автоматически созданное описание

Описание тестового покрытия



Для тестирования мы использовали табличные значения для проверки тригонометрических функций и пограничные значения области определения для логарифмических функций. Кроме того, использовались нечисловые значения, к примеру +\- бесконечности.

График по CSV – выгрузкам



Исходный код



<https://github.com/EgorMIt/TpoLab2>

Вывод

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.