

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«Санкт-Петербургский национальный исследовательский университет
информационных технологий, механики и оптики»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И
КОМПЬЮТЕРНОЙ ТЕХНИКИ

ОТЧЁТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №2
по дисциплине
«Тестирование программного обеспечения»
Вариант №4616

Выполнил:
Студент группы Р3314
Минкова Алина Андреевна
Преподаватель:
Харитоновна Анастасия Евгеньевна

ИТМО

Санкт-Петербург, 2025

Оглавление

Задание	3
UML-диаграмма классов разработанного приложения	4
Описание тестового покрытия	5
Графики, построенные csv-выгрузкам, полученные в процессе интеграции приложения	7
Вывод.....	8

Задание

Лабораторная работа #2

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

Введите вариант:

$$\begin{cases} \left(\left(\left(\cot(x) \cdot \cos(x) \right) + \sin(x) \right) - \tan(x) \right) - \left(\cot(x) \cdot \sin(x) \right) + \left(\cos(x)^3 \right) & \text{if } x \leq 0 \\ \left(\left(\left(\frac{\log_5(x) \cdot \log_{10}(x)}{\log_2(x)} \right) - ((\ln(x) \cdot \ln(x)) + \log_{10}(x)) \right) + \log_{10}(x) \right) & \text{if } x > 0 \end{cases}$$

$x \leq 0 : (((\cot(x) * \cos(x)) + \sin(x)) - \tan(x)) - (\cot(x) * \sin(x))) + (\cos(x) ^ 3))$

$x > 0 : (((((\log_5(x) * \log_{10}(x)) / \log_2(x)) - ((\ln(x) * \ln(x)) + \log_{10}(x))) + \log_{10}(x)))$

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 $\sin(x)$):

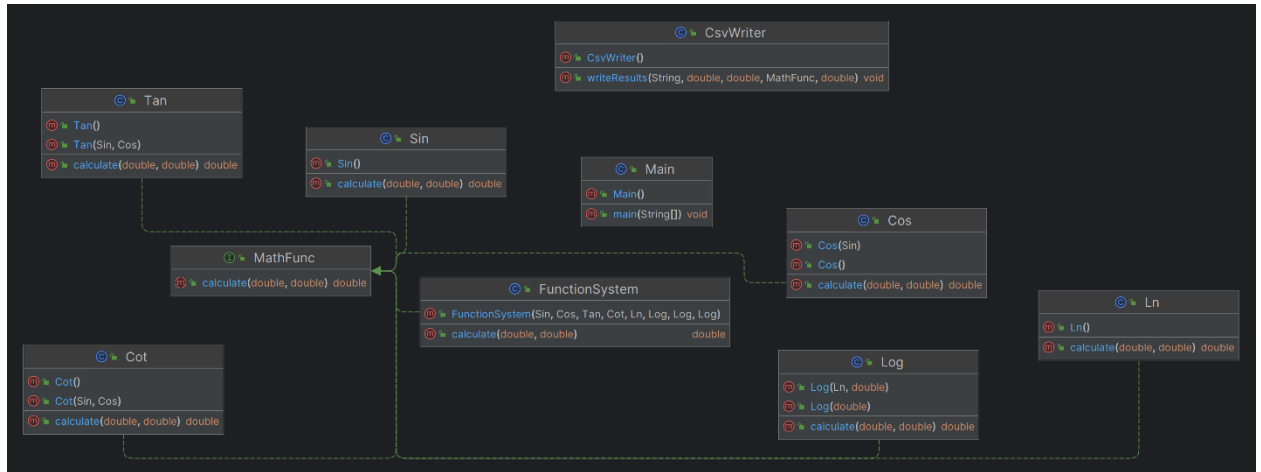


3. Обе "базовые" функции (в примере выше - $\sin(x)$ и $\ln(x)$) должны быть реализованы при помощи разложения в ряд с задаваемой погрешностью. Использовать тригонометрические / логарифмические преобразования для упрощения функций ЗАПРЕЩЕНО.
4. Для КАЖДОГО модуля должны быть реализованы табличные заглушки. 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, 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 разработанного приложения



Исходный код:

https://github.com/aulouu/tpo_lab2

Описание тестового покрытия

Тестовое покрытие для данной системы включает модульные и интеграционные тесты, проверяющие корректность работы математических функций и их взаимодействие.

Модульные тесты:

1. Тестирование функции Sin:
 - testSinZero(): Проверяет вычисление синуса для 0 (ожидается 0.0)
 - testSinPiDivTwo(): Проверяет вычисление синуса для $\pi/2$ (ожидается 1.0)
 - testSinPi(): Проверяет вычисление синуса для π (ожидается 0.0)
 - testSinInvalidEpsilon(): Проверяет обработку недопустимого значения эпсилон (отрицательное)
2. Тестирование функции Cos (с моком Sin):
 - testCosUsingSin(): Проверяет вычисление косинуса через синус ($\cos(x) = \sin(\pi/2 - x)$)
3. Тестирование функции Tan (с моками Sin и Cos):
 - testCotUsingSinAndCos(): Проверяет вычисление тангенса как отношение \sin/\cos
4. Тестирование функции Cot (с моками Sin и Cos):
 - testCotUsingSinAndCos(): Проверяет вычисление котангенса как отношение \cos/\sin
5. Тестирование функции Ln:
 - testLnOne(): Проверяет $\ln(1) = 0$
 - testLnE(): Проверяет $\ln(e) = 1$
 - testLnLessThanOne(): Проверяет вычисление для значений < 1
 - testLnGreaterThanOne(): Проверяет вычисление для значений > 1
 - testLnZero(): Проверяет обработку нулевого аргумента
 - testLnNegative(): Проверяет обработку отрицательного аргумента
 - testLnInvalidEpsilon(): Проверяет обработку недопустимого эпсилон
6. Тестирование функции Log (с моком Ln):
 - testLogUsingLn(): Проверяет вычисление логарифма через натуральный логарифм

Интеграционные тесты:

1. Тестирование FunctionSystem для $x \leq 0$:
 - Проверяет комбинацию тригонометрических функций

- Использует моки для всех зависимостей
 - Проверяет корректность вычисления составного выражения
2. Тестирование FunctionSystem для $x > 0$:
- Проверяет комбинацию логарифмических функций
 - Использует моки для всех зависимостей
 - Проверяет корректность вычисления составного выражения

Тесты покрывают:

- Все основные математические функции (sin, cos, tan, cot, ln, log)
- Граничные случаи (нулевые и отрицательные значения)
- Обработку ошибок (недопустимые параметры)
- Взаимодействие между функциями в системе
- Разделение поведения для положительных и отрицательных x

Графики, построенные csv-выгрузкам, полученные в процессе интеграции приложения

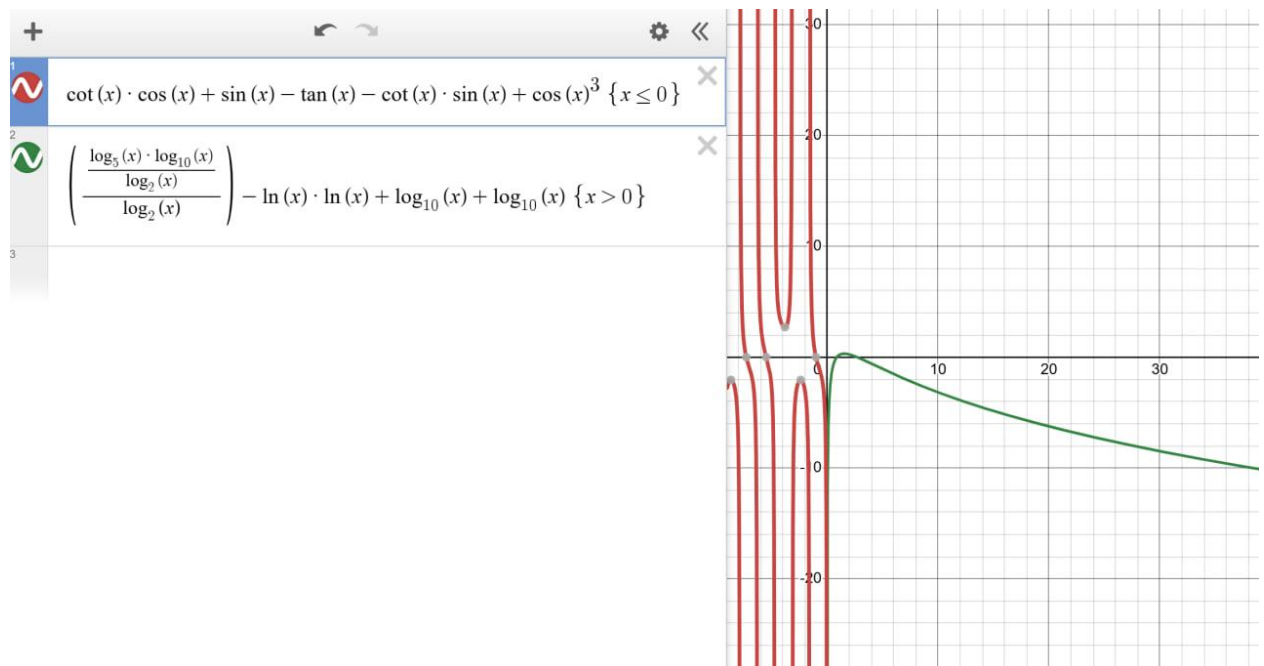
Исходная функция:

Введите вариант:

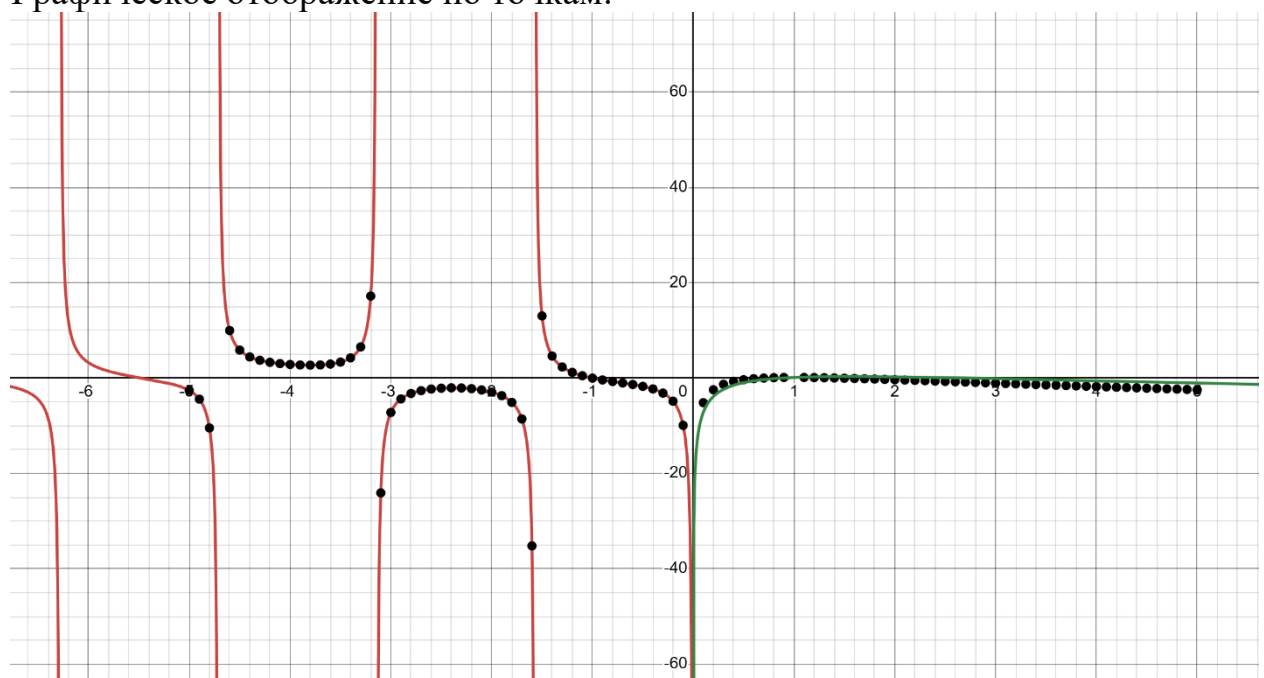
$$\begin{cases} \left(\left(\left(\left(\cot(x) \cdot \cos(x) \right) + \sin(x) \right) - \tan(x) \right) - \left(\cot(x) \cdot \sin(x) \right) \right) + \left(\cos(x)^3 \right) & \text{if } x \leq 0 \\ \left(\left(\left(\frac{\log_5(x) \cdot \log_{10}(x)}{\log_2(x)} \right) - \left(\ln(x) \cdot \ln(x) \right) + \log_{10}(x) \right) \right) + \log_{10}(x) & \text{if } x > 0 \end{cases}$$

$x \leq 0 : (((((\cot(x) * \cos(x)) + \sin(x)) - \tan(x)) - (\cot(x) * \sin(x))) + (\cos(x) ^ 3))$

$x > 0 : (((((\log_5(x) * \log_{10}(x)) / \log_2(x)) / \log_2(x)) - ((\ln(x) * \ln(x)) + \log_{10}(x))) + \log_{10}(x))$



Графическое отображение по точкам:



Вывод

В ходе выполнения лабораторной работы было разработано приложение, реализующее систему математических функций (тригонометрических и логарифмических) через базовые разложения в ряд ($\sin(x)$ и $\ln(x)$), с соблюдением требований: запрета использования упрощающих преобразований, создания табличных заглушек для каждого модуля, обработки областей допустимых значений и взаимозависимых точек, а также поддержки вывода результатов в CSV-файл с настраиваемым шагом. Тестовое покрытие на JUnit4, включающее модульные и интеграционные тесты с анализом эквивалентности, подтвердило корректность работы системы как отдельных компонентов, так и их взаимодействия, что было верифицировано поэтапной интеграцией модулей с контролем покрытия. Результаты демонстрируют соответствие реализации заданным правилам и надежность системы при обработке различных входных данных.