Практическая работа №5

«Отладка отдельных модулей программного проекта»

Выполнила: Юсупова Камола

**Цель работы**

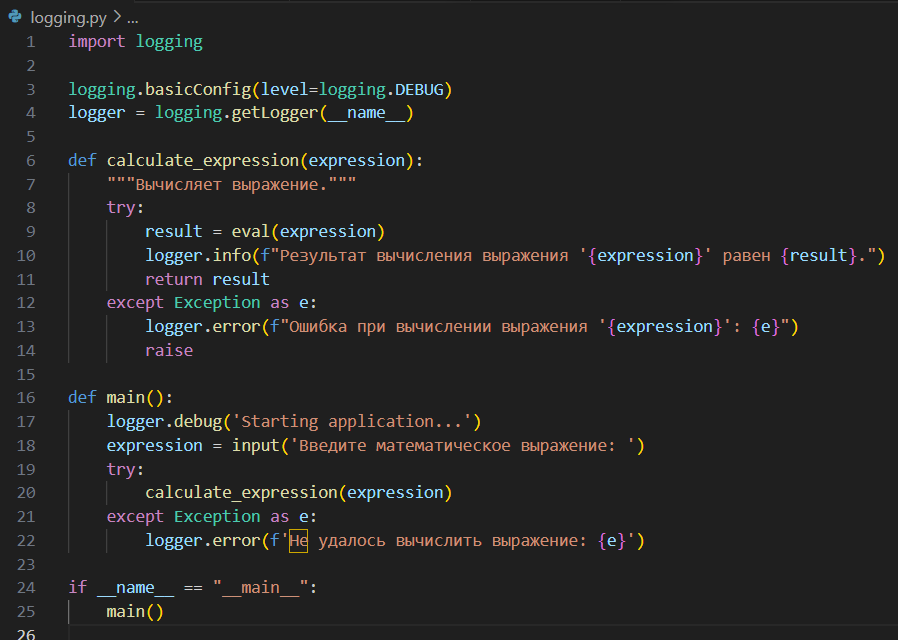
Цель данной работы – изучить процесс отладки отдельных модулей программного проекта, выявить типичные ошибки и научиться эффективно использовать инструменты отладки для их устранения.

**Задание 1: Подготовка среды для отладки**

1. Настроила среду разработки для работы с проектом.

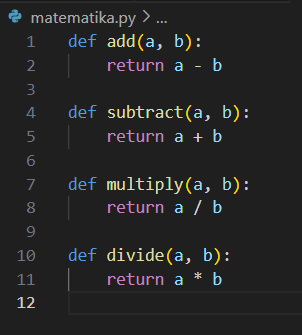
В Visual Studio Code стандартным инструментом отладки для Python является встроенная поддержка отладки, которая предоставляется расширением Python от Microsoft. Чтобы начать использовать этот инструмент, я установила соответствующее расширение. Создала виртуальное окружение с помощью кода python -m venv .venv. Создала конфигурацию запуска.

1. Убедилась, что отладчик корректно подключен к проекту, и что можно запускать отладку по шагам.
2. Настроила логирование для отслеживания ошибок.

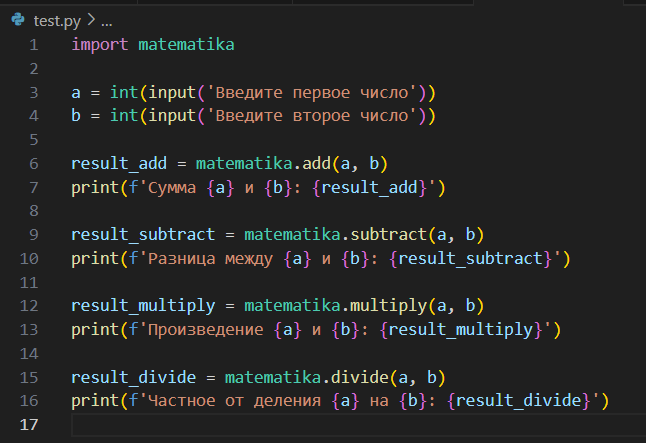


**Задание 2: Отладка логики модуля**

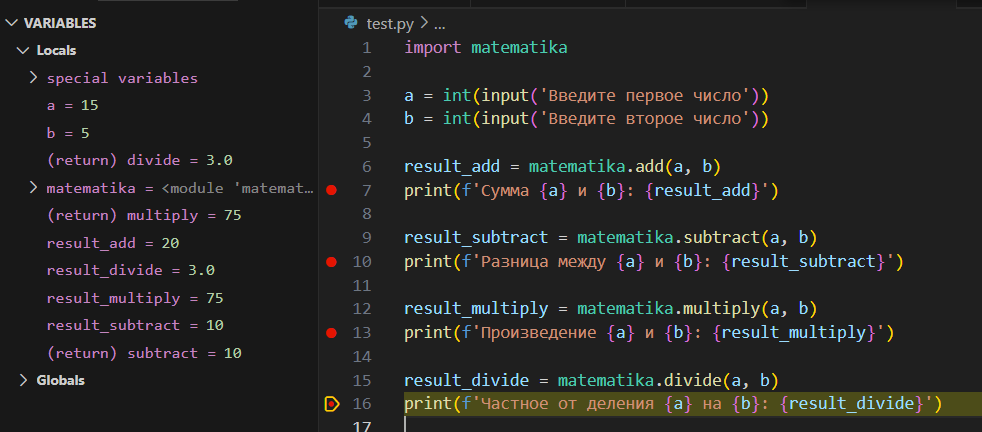
1. Выбрала один из модулей программного проекта, который требует отладки.



1. Используя пошаговую отладку, выявила ошибки в логике работы модуля.

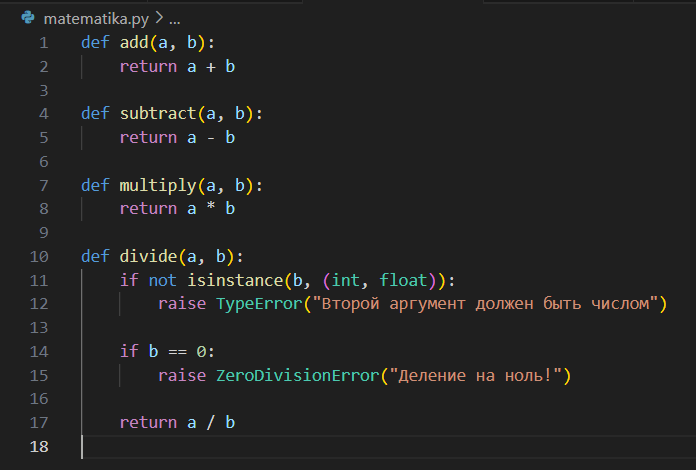


1. Протестировала корректность входных данных и отлов исключений.



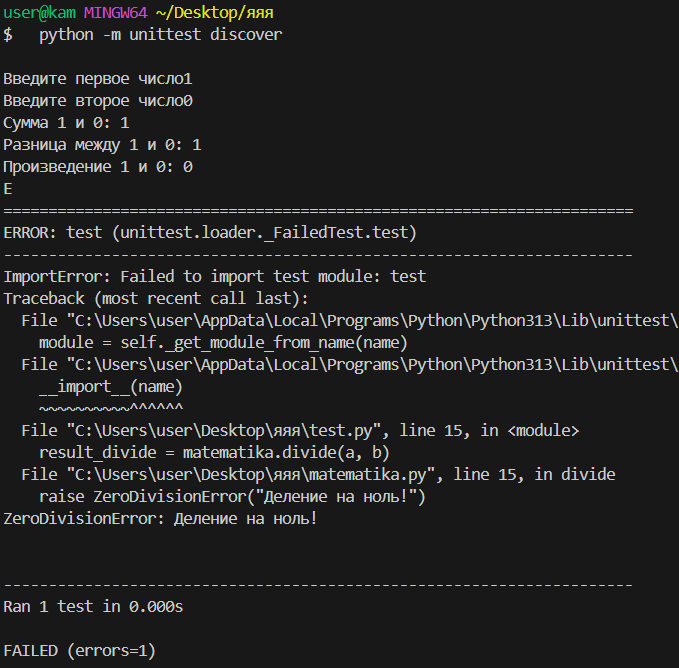
1. Задокументировала выявленные ошибки и предложила способы их исправления.

Входные данные для функции divide могут быть некорректного типа, что вызывает ошибку TypeError. Также может произойти ошибка ZeroDivisionError.

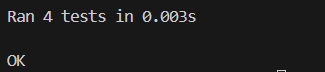


**Задание 3: Интеграционная отладка**

1. Провела тестирование на корректную работу модуля после интеграции с другими частями проекта.



1. Используя отладчик, проследите работу модуля в рамках всего проекта, убедитесь в отсутствии конфликтов с другими модулями.

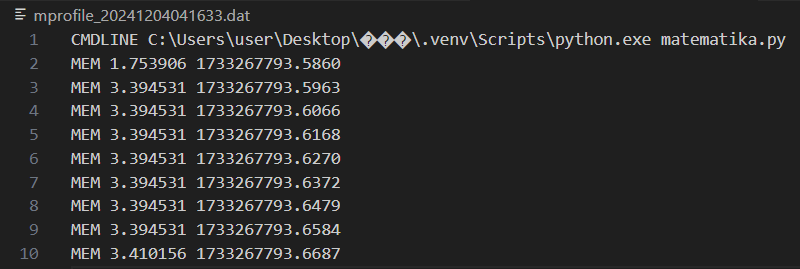


1. Исправила найденные ошибки, обновила документацию.

**Задание 4: Отладка производительности и использования памяти**

1. Используя инструменты анализа производительности, проанализировала использование памяти и время выполнения модуля.

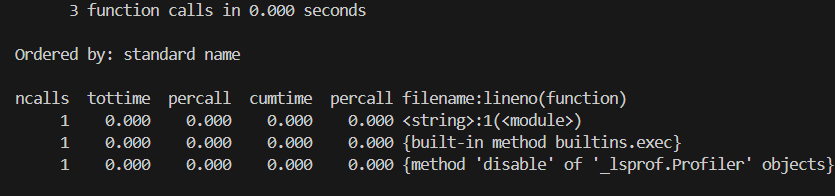




1. Выявила участки кода, которые требуют оптимизации.

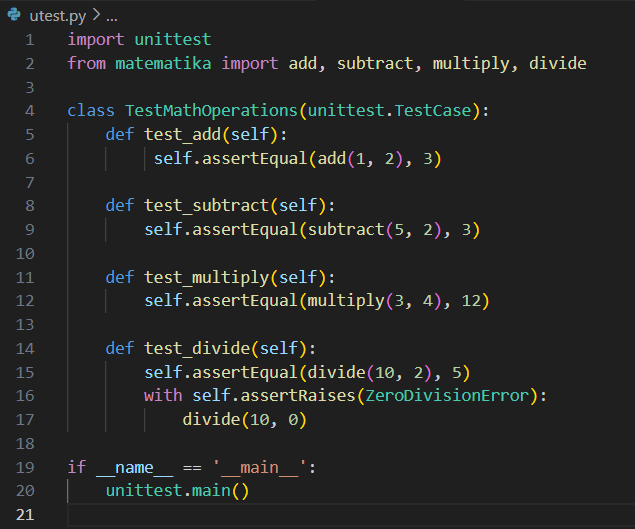


1. Предложила способы оптимизации кода и реализовала их.
2. Протестировала модуль после оптимизаций, чтобы убедиться в улучшении производительности.

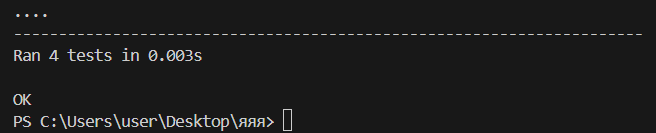


**Задание 5: Автоматизированное тестирование и отладка**

1. Разработала автоматизированные тесты для отладки выбранного модуля.



1. Используя тесты, провела регрессионное тестирование для проверки стабильности работы модуля после внесения изменений.



1. Исправила все выявленные ошибки в процессе автоматического тестирования и повторила тестирование.

**Задание 6: Отладка проблем с параллелизмом**

В модуле проекта не используются многопоточность и асинхронное программирование. Но для их предотвращения можно использовать механизмы синхронизации, такие как threading.Lock и asyncio.Lock, которые обеспечивают безопасный доступ к разделяемым ресурсам.

**Вывод**

Изучила процесс отладки отдельных модулей программного проекта, выявила типичные ошибки и научилась эффективно использовать инструменты отладки для их устранения.