

Лабораторная работа №1. Разработка консольного калькулятора на Python

Цель работы

Разработка программы для вычисления математических выражений с поддержкой основных арифметических операций и унарных знаков.

Задачи работы

1. Реализовать разбор математических выражений на токены
2. Обеспечить обработку унарных плюса и минуса
3. Реализовать вычисления с учётом приоритета операций
4. Организовать код в модульной структуре
5. Реализовать обработку ошибок
6. Написать модульные тесты

Требования к программе

- Ввод: строка с математическим выражением без скобок
- Поддерживаемые операции
- Приоритет операций: умножение и деление выше сложения и вычитания
- Поддержка унарных знаков
- Формат чисел: целые и вещественные
- Обработка ошибок: деление на ноль, неверные символы

Структура программы

Программа состоит из трёх основных модулей:

1. ****tokenizer.py**** - модуль токенизации
 - Функция `get_tokens()` разбивает строку на токены
 - Поддерживает унарные знаки и вещественные числа
 - Выявляет недопустимые символы
2. ****calculator.py**** - модуль вычислений
 - Функция `calculate()` вычисляет значение выражения
 - Реализует приоритет операций
 - Обрабатывает деление на ноль
3. ****main.py**** - главный модуль
 - Функция `main()` обеспечивает взаимодействие с пользователем
 - Организует ввод-вывод данных
 - Обрабатывает исключения

Алгоритм работы

Этап 1: Токенизация

Программа принимает строку выражения и разбивает её на токены - числа и операторы. Унарные знаки обрабатываются как часть чисел.

Этап 2: Вычисление

Выполняется два линейных прохода по токенам:

1. Обработка умножения и деления
2. Обработка сложения и вычитания

Тестирование

Написаны модульные тесты с использованием pytest:

- Тесты токенизации: проверка разбора выражений, унарных знаков, обработки ошибок
- Тесты вычислений: проверка операций, приоритета, обработки деления на ноль

Все тесты успешно проходят, покрывая основные сценарии работы программы.

Результаты работы

Разработана консольная программа-калькулятор, которая:

- Корректно разбирает математические выражения
- Правильно обрабатывает унарные знаки
- Соблюдает приоритет операций
- Обрабатывает ошибки ввода
- Имеет модульную структуру
- Протестирована на различных сценариях

Программа демонстрирует практическое применение основ Python: работу со строками, функциями, модулями, обработкой исключений и тестированием.