

# Лабораторная работа № 3 по курсу дискретного анализа: Профилирование

Выполнил студент группы 08-215 МАИ *Тараскаев Давид*.

## Дневник отладки

Проведено нагрузочное тестирование на 10 000 000 операций с последующим анализом результатов через `gprof` и `valgrind`.

## Инструмент `gprof`

### До оптимизации:

- Функция `to_lower` занимала **16,6%** общего времени.
- Общее число вызовов — **13 331 642**.
- Причина: при каждом поиске ключа строка приводилась к нижнему регистру заново.

### После оптимизации:

- Доля времени на `to_lower` снизилась до **11,55%**.
- Количество вызовов упало до ровно **10 000 000**, то есть количество вызовов стало равно количеству операций в тестовых данных.
- Оптимизация: преобразование ключа к нижнему регистру выполняется единожды перед вставкой, поиском или удалением; в узлах хранятся уже «нижнерегистровые» ключи.

## Инструмент `valgrind`

### До оптимизации:

- Программа имела утечки по памяти.
- Причина: не был указан деструктор класса.

### После оптимизации:

- После добавления деструктора у класса, утечки по памяти пропали.

## Тест производительности

До оптимизации в реализованном AVL-дереве выполнение 10 миллионов случайных команд заняло 28.52 секунды, а после оптимизации — 28.35 секунды. Разница во времени выполнения незначительна на данном объёме данных, однако при работе с большими объёмами она может стать более заметной.

## Выводы

Профилировщики gprof и valgrind являются мощными и удобными инструментами для анализа производительности и выявления проблем в программе. Несмотря на простоту их использования, они выполняют важнейшую работу: позволяют обнаружить "узкие места" утечки памяти и избыточные вычисления, что особенно ценно при оптимизации сложных структур данных.