

# Анализ влияния кэша на операцию матричного умножения.

## Задание

**Задача:** Реализовать последовательный алгоритм матричного умножения и оценить влияние кэша на время выполнения программы. На основе анализа влияния кэша построить графики, составить отчет.

**Формат файла-матрицы:** Матрица представляются в виде бинарного файла следующего формата:

| Тип                 | Значение                      | Описание                     |
|---------------------|-------------------------------|------------------------------|
| char                | T - d(int32_t) или l(int64_t) | Тип элементов                |
| int32_t             | N > 0                         | Число строк/столбцов матрицы |
| Массив чисел типа T | N x N элементов               | Массив элементов матрицы     |

Элементы матрицы хранятся построчно. Матрица квадратная.

**Формат командной строки:** <binary> <матрица\_a> <матрица\_b> <матрица\_c> <режим> Режимы: 0 – ijk, 1 – ikj, 2 – kij, 3 – jik, 4 – jki, 5 – kji.

**Пример запуска:**

```
$ ./run a b c 0
```

## Требования к решению

Код должен компилироваться gcc v7.2.0 с опциями компиляции -Wall -Werror -O0.

Программа должна корректно отрабатывать при компиляции с опцией -fsanitize=address.

Решение должно содержать Makefile.

Makefile должен содержать target test.

При вызове make test должен запускаться скрипт ./test.sh.

Бинарник, передаваемый в ./test.sh должен быть скомпилирован с опцией -fsanitize=address.

Для оценки производительности полученного решения, опцию -fsanitize=address следует отключить.

Решение нужно прислать через Pull Request, все изменения должны быть внутри директории `./problem1`.