



**Спецкурс: системы и средства параллельного
программирования.**

Отчёт № 1.

**Анализ влияния кэша на операцию матричного
умножения.**

Работу выполнил
Тимачев А. А.

Москва 2018

Постановка задачи и формат данных.

Задача: Реализовать последовательный алгоритм матричного умножения и оценить влияние кэша на время выполнения программы.

Формат командной строки: <имя файла матрицы A > <имя файла матрицы B > <имя файла матрицы C > <режим, порядок индексов>.

Режимы: 0 – ijk, 1 – ikj, 2 – kij, 3 – jik, 4 – jki, 5 – kji.

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

Тип	Значение	Описание
Число типа char	T – f (float) или d (double)	Тип элементов
Число типа uint64_t	N – натуральное число	Число строк матрицы
Число типа uint64_t	M – натуральное число	Число столбцов матрицы
Массив чисел типа T	$N \times M$ элементов	Массив элементов матрицы

Элементы матрицы хранятся построчно.

Описание алгоритма.

Математическая постановка: Алгоритм матричного умножения ($A \times B = C$) можно представить в следующем виде: $c_{ij} = \sum_k a_{ik} \cdot b_{kj}$ для каждого элемента матрицы C . Оценка влияния кэша на время выполнения программы осуществляется за счёт перестановки индексов суммирования.

Анализ времени выполнения: Для оценки времени выполнения программы использовалась библиотека chrono. Для повышения надёжности экспериментов опыты проводились несколько раз (10).

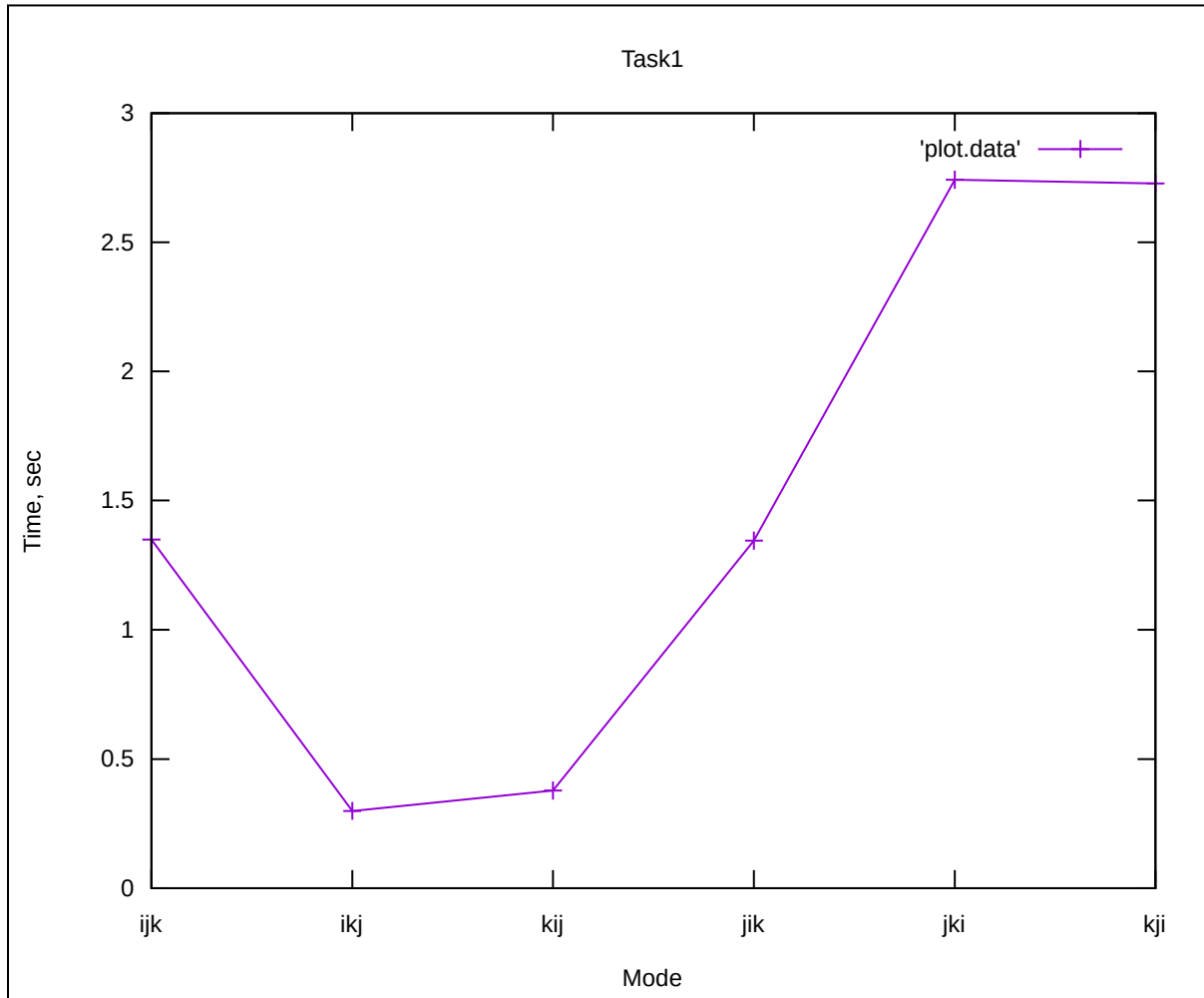
Верификация: Для проверки корректности работы программы использовались тестовые данные.

Основные функции:

- **Чтение матрицы из файла.** В рамках функции осуществляется чтение параметров матрицы и значений ее элементов.
- **Перемножение матриц.** В рамках функции осуществляется перемножение матриц в соответствие с выбранным порядком индексов суммирования.
- **Запись матрицы в файл.** В рамках функции осуществляется запись параметров матрицы и значений ее элементов.

Результаты выполнения.

Результаты:



Проводилось перемножение двух матриц с размерами 500x500. Зависимость времени выполнения от порядка индексов суммирования представлена на графике (время в секундах).

Основные выводы.

Исследования показывают, что изменения порядка индексов суммирования оказывает влияние на время выполнения программы. Наименьшее время выполнения при следующих порядках индексов — ikj и kij . При таких порядках доступ к элементам обеих входных матриц осуществляется последовательно. Наихудшее время при порядках jki и kji . При таких подходах доступ к памяти осуществляется максимально непоследовательно.