

Задание 4

Отчёт

по исследованию

высокопроизводительного

умножения матриц на основе

SIMD

Ши Хуэй shihuicollapsor@gmail.com

1. Постановка задачи

Целью данного отчета является исследование высокопроизводительного умножения матриц с использованием технологий SIMD (Single Instruction, Multiple Data). В рамках работы предполагается реализация и сравнение различных алгоритмов умножения матриц, включая последовательный алгоритм, а также векторизованные версии, использующие расширения AVX для архитектуры x86 и VSX для архитектуры IBM POWER.

1. Изучение теории SIMD: Анализ принципов работы SIMD и его применения в высокопроизводительных вычислениях.
2. Реализация алгоритмов умножения матриц: Разработка последовательного алгоритма, а также векторизованных реализаций с использованием AVX и VSX.
3. Сравнение производительности: Проведение тестов для оценки времени выполнения различных реализаций алгоритмов на матрицах различных размеров ($N = 512, 1024, 2048$) и сравнение производительности AVX и VSX.

Таким образом, задача состоит в том, чтобы продемонстрировать преимущества использования SIMD для ускорения операций с матрицами и выявить различия в производительности между AVX и VSX.

2. Формат командной строки

```
g++ -o work4 work4.cpp -mavx
```

3. Спецификация системы

- Architecture: x86_64
- CPU op-mode(s): 32-bit, 64-bit
- Address sizes: 39 bits physical, 48 bits virtual
- Byte Order: Little Endian
- CPU(s): 12

- On-line CPU(s) list: 0-11
- Vendor ID: GenuineIntel
- Model name: Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz
- CPU family: 6
- Model: 165
- Thread(s) per core: 2
- Core(s) per socket: 6
- Socket(s): 1
- Stepping: 2
- CPU(s) scaling MHz: 26%

5. Записи экспериментов и результаты

Результаты для AVX:

N = 512

Sequential version time: 0.896500 seconds

AVX version time: 0.479271 seconds

Results are correct!

N = 1024

Sequential version time: 7.535754 seconds

AVX version time: 5.151238 seconds

Results are correct!

N = 2048

Sequential version time: 246.413924 seconds

AVX version time: 147.557310 seconds

Results are correct!