

Реализовать параллельный алгоритм матричного умножения.

Задание

Задача: Реализовать параллельный 3D алгоритм DNS матричного умножения (разработать параллельную программу с использованием технологии MPI). Протестировать полученную программу на Bluegene. Составить графики времени работы, ускорения, эффективности для матриц размерности 512×512 , 1024×1024 , 1536×1536 , 2048×2048 и для числа процессов:

- 1;
- $8 = 2^3$;
- $64 = 4^3$;
- в качестве бонуса: $512 = 8^3$.

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

Тип	Значение	Описание
int32_t	$N > 0$	Число строк/столбцов матрицы
Массив чисел типа int32_t	$N \times N$ элементов	Массив элементов матрицы

Элементы матрицы хранятся построчно. Матрица квадратная.

Формат командной строки: `<binary> <матрица_a> <матрица_b> <матрица_c>`

Пример запуска:

```
$ mpirun -n 64 ./run -- a b c
```

Требования к решению

Можно использовать пользовательские типы данных MPI (user-defined datatypes) для распределения матрицы между узлами и для сбора результата для последующего сохранения их в файле.

Ни один узел (кроме, возможно, мастер-узла для ввода/вывода) не должен хранить всю матрицу целиком.

Гарантируется, что число MPI-процессов является кубом натурального числа: $N_p = p^3$, а размер матрицы $n \times n$ таков, что n кратно p .

Решение должно содержать Makefile.

Makefile должен содержать target test.

При вызове `make test` должен запускаться скрипт `./test.sh`¹.

¹Тесты не будут работать на bluegene, потому что там используется BigEndian.