

Лабораторная работа 8.

Задание 1. Параллелизация вычислений

Общая часть:

На основе класса матрицы из лабораторной работы №2 разработать шаблон.

Доработать интерфейс таким образом, чтобы он содержал методы для записи и считывания матрицы (как целиком, так и только данных) в/из файла.

Формат входных данных:

N M

строка1столбец1 строка1столбец2 ... строкаNстолбецM

, где N – число строк, M – число столбцов.

Реализовать конструкторы, позволяющие создавать матрицы указанного размера, а также считывающие матрицы из файла.

Перегрузить операторы умножения матриц, а также матриц и скаляров.

Перегрузить операторы сложения/вычитания матриц.

Перегрузить оператор логического отрицания для вычисления обратной матрицы (если обратная матрица не может быть вычислена следует запускать исключение).

Перегрузить операторы проверки на равенство для матриц.

Перегрузить операторы проверки на равенство для матриц и целых значений 0, а также 1, проверяющих что матрица нулевая или единичная.

Перегрузить оператор присваивания.

Добавить статические методы создания нулевых и единичных матриц указанного размера.

Задание 1 (6 баллов). Параллельные потоки std::thread.

Реализовать параллельные версии алгоритмов из общей части.

Задание 2* (2 балла). Параллельные потоки std::async.

Перегрузить параллельные версии алгоритмов из задания 1 таким образом, чтобы они выполняли блочную обработку: принимали размер блока, выполняли обработку и возвращали результат используя фьючерс.

Задание 3* (2 балла). Применение Win32-API/POSIX Threads.

Выполнить задание 1 используя стандарт Си++98 и возможности Win32-API либо POSIX Threads

Список литературы:

1. Энтони Уильямс «Параллельное программирование на C++ в действии. Практика разработки многопоточных программ»
2. Майерс Скотт «Эффективный и современный C++»
3. Джосьюттис Н. «C++. Стандартная библиотека»