Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

**«Самарский НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ**

**имени академика С.П. Королева»**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

ПАРАЛЛЕЛЬНОЕ ПРОГРАММИРОВАНИЕ

**Отчёт по лабораторной работе №3**

ПРОГРАММА ПЕРЕМНОЖЕНИЯ ДВУХ МАТРИЦ ПО MPI

Жирнов Владислав

Группа 6313-100503D

Самара 2025

# **Цель работы**

Модифицировать программу для параллельной работы по технологии MPI.

# **2 Программа перемножения двух матриц на языке C++**

Программа умножения двух матриц на языке C++ была модифицирована для поддержки параллельных вычислений с использованием технологии MPI. Исходный код (main.cpp) реализует алгоритм перемножения квадратных матриц с измерением времени выполнения операций.

Основной акцент был сделан на эффективное распределение вычислительной нагрузки между несколькими процессами.

Создается 11 пар квадратных матриц с размерами от 10x10 до 100x100 с шагом 10, а также матрица с размером 1000x1000. Каждая пара матриц одного размера сохраняется в соответствующих директориях: Matrix\_1 содержит левые множители, а Matrix\_2 - правые множители. Таким образом, после завершения генерации в каждой из этих папок оказывается по 10 матриц соответствующих размеров - от 10x10 до 100x100.

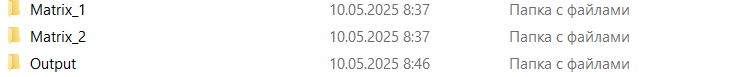


Рисунок 1 – основные каталоги, используемые при работе программы

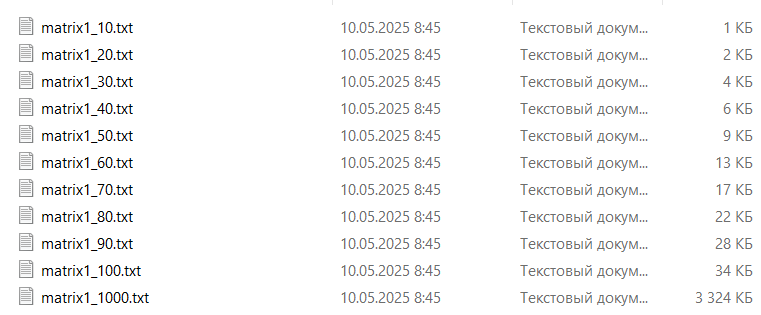
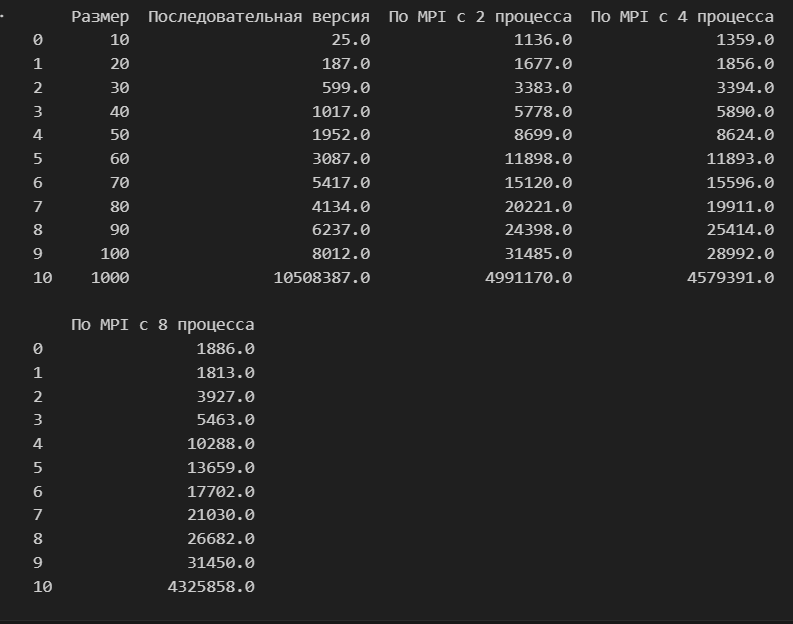


Рисунок 2 – внутренние каталоги каталога Matrix\_1.

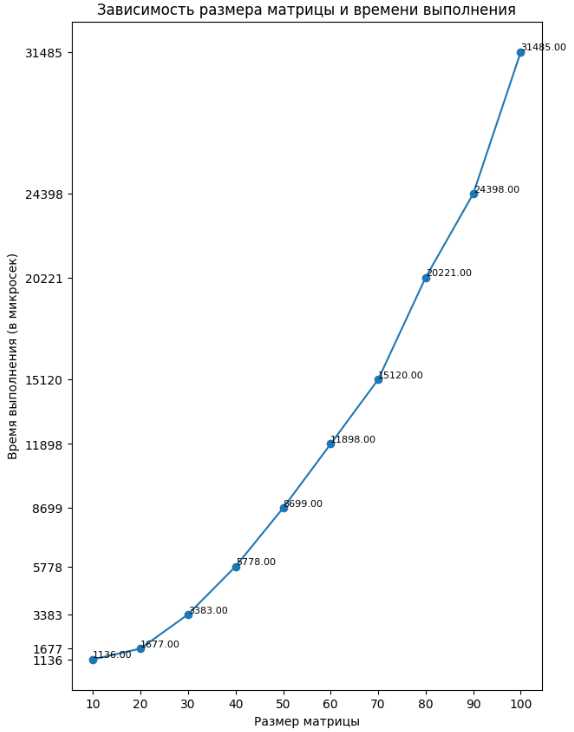
Исходные матрицы считываются из соответствующих файлов в директориях Matrix\_1 и Matrix\_2, после чего производится их поэлементное перемножение. Программа сохраняет время выполнения операций для каждого размера матрицы в файл time\_mpi\_N.txt, где N - количество используемых процессов.

Программа демонстрирует практическое применение технологии MPI для решения вычислительно сложных задач линейной алгебры, обеспечивая значительное ускорение вычислений за счет параллельной обработки данных.

# **Экспериментальные результаты**

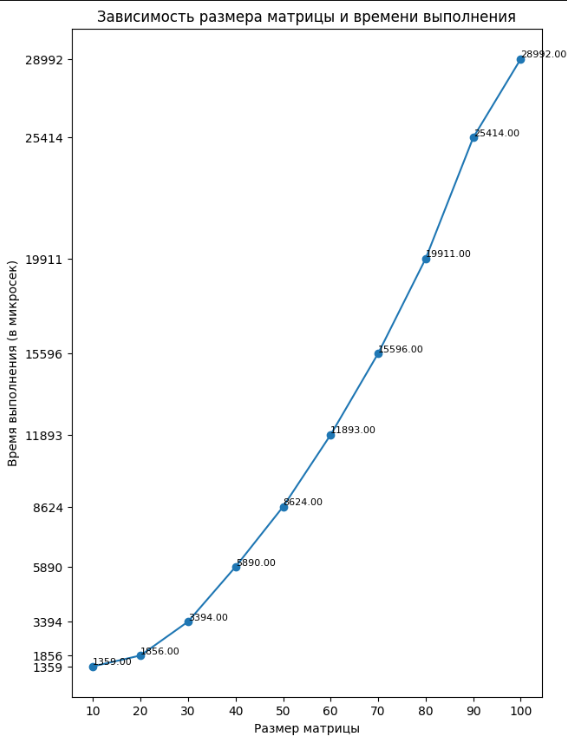


MPI с 2 процессами:



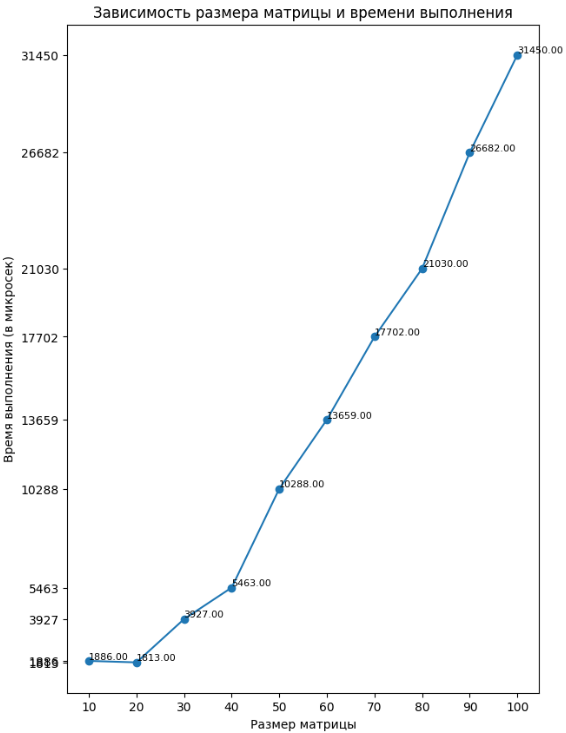
Время для матрицы 1000x1000: 4991170

MPI с 4 процессами:



Время для матрицы 1000x1000: 4579391

MPI с 8 процессами:



Время для матрицы 1000x1000: 4325858

Вывод: Наблюдаемая линейная зависимость наглядно демонстрирует временную сложность операции умножения матриц. Это подтверждает теоретические ожидания - с увеличением размерности матриц время вычислений растёт пропорционально. Однако использование MPI-параллелизации позволяет значительно сократить общее время выполнения операций за счёт распределения вычислений между несколькими процессами. Таким образом, для работы с матрицами больших размеров применение MPI становится особенно целесообразным, так как обеспечивает существенную экономию вычислительного времени по сравнению с последовательными алгоритмами.