Федеральное государственное автономное образовательное учреждение высшего

образования «ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ» Факультет математики, механики и компьютерных наук Кафедра математического моделирования Направление подготовки 010501 —«Прикладная математика и информатика»

**Индивидуальная работа на тему: «Параллельное блочное умножение матриц»**

Студентки 4 курса: М. С. Редун

Ростов-на-Дону 2018

**Отчет по индивидуальному заданию «Блочное умножение матриц»**

**Вариант 41.**

**Постановка задачи.**

Написать программу блочного умножения двух матриц C = A\*B. Матрица A верхне-треугольная. Хранится в виде одномерного массива по блочным строкам. Матрица B симметричная, хранится как нижне-треугольная. Хранится в виде одномерного массива по блочным строкам. Распараллелить блочную программу умножения двух матриц C = A\*B с использованием технологии OpenMP двумя способами • Перемножение каждых двух блоков выполнить параллельно • В разных вычислительных ядрах одновременно перемножать разные пары блоков. Определить оптимальные размеры блоков в обоих случаях. Провести численные эксперименты и построить таблицу сравнений времени выполнения различных программных реализаций решения задачи. Определить лучшие реализации. Проверить корректность (правильность) программ.

**Описание программной реализации и результатов исполнения:**

В данной работе представлен блочный механизм умножения матриц для последовательного и параллельного случаев. Для упрощения вычислений выбираются квадратные матрицы n × n с размером блока, равного степени числа 2.

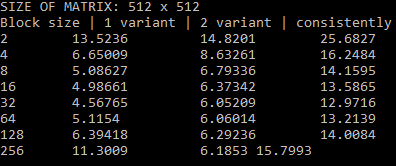
Все рассмотренные методы характеризуются одинаковой вычислительной трудоемкостью и равным объемом передаваемых данных. Однако параллельные алгоритмы являются более эффективными за счет возможности равномерного распределения вычислительной нагрузки между процессорами. В отличие от последовательного, с операциями сложения и умножения, исполняющимися в 1 процессоре. Каждый из потоков отвечает за вычисление одного блока результирующей матрицы С. Задача разделяется на отдельные подзадачи, связанные с перемножением пары блоков исходных матриц (А-верхне-треугольной, В-симметричной, нижне-треугольной) и равные заданному числу потоков-4. Для 1 параллельного случая. 2 вариант позволяет контролировать число потоков за счет распределения вычислений между ядрами процессора.

Так, тестирование программы проводилось на компьютере со следующей конфигурацией:

ПРОЦЕССОР Intel Core i5 4200H ( 2 физических, 4 логических ядра, частота-2.8 Ггц), ОПЕРАТИВНАЯ ПАМЯТЬ 8Gb, ОПЕРАЦИОННАЯ СИСТЕМА Windows 10 Pro x64. Программные средства: компилятор С++ MSVC\_2017\_64 для фреймворка Qt.

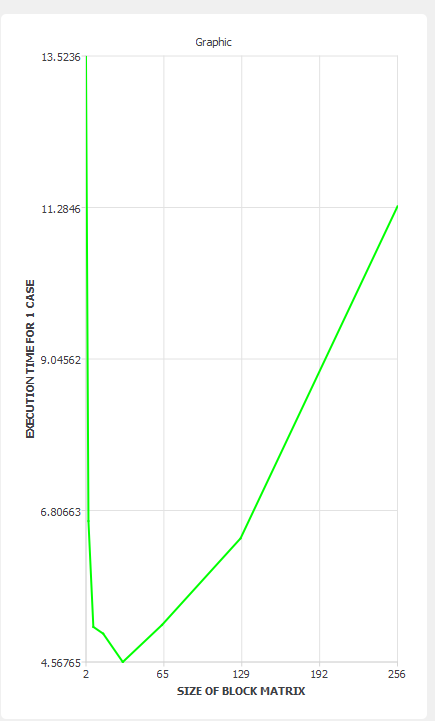
На данном изображении представлена таблица времени выполнения различных программных реализаций задачи. Если начальную матрицу выбрать порядка 1024 × 1024, то в качестве оптимального размера блока для всех случаев можно считать подматрицy: 32 × 32

**Рис-1.Таблица сравнений времени выполнения для разных случаев**

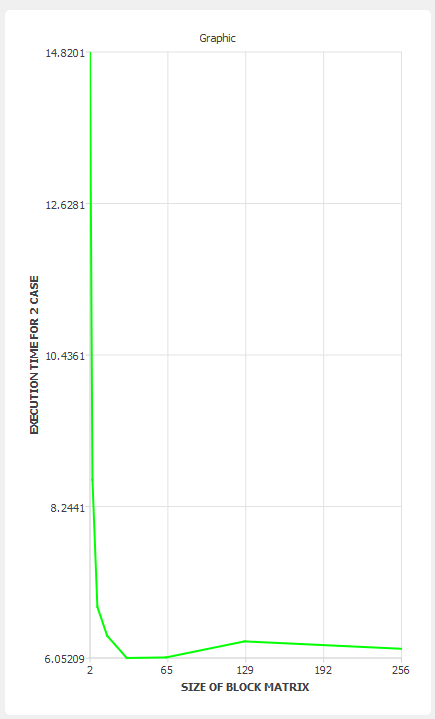
****

**Рис-2.Графики зависимости времени от размера блока**

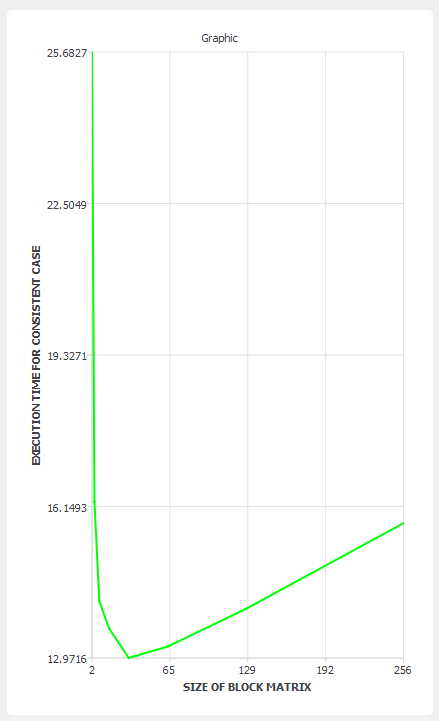
Параллельный случай 1



Параллельный случай 2



Последовательный случай



Снимки выше иллюстрируют графическое представление зависимости времени исполнения программы от размера блока. Лучшая реализация изображена на 1 рисунке, так как содержит минимальное время исполнения алгоритма. А время на вычисления для параллельных случаев по сравнению с последовательным затрачивается в 2 раза меньше при выборе любого из блоков.