**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра МО ЭВМ**

**ОТЧЕТ**

**по лабораторной работе №1**

**по дисциплине «Параллельные алгоритмы»**

**Тема: Основы работы с процессами и потоками**

| Студент гр. 9304 |  | Арутюнян С.Н. |
| --- | --- | --- |
| Преподаватель |  | Сергеева Е.И. |

Санкт-Петербург

2022

## Цель работы.

Ознакомиться с работой с процессами и потоками в языке программирования C++.

**Задание.**

Выполнить поэлементное сложение 2х матриц M\*N

Входные данные: Две матрицы вводятся из файла или генерируются.

Результат: Сумма двух матриц записывается в файл

1. Выполнить задачу, разбив её на 3 процесса. Выбрать механизм обмена данными между процессами.
   1. Процесс 1: заполняет данными входные матрицы (читает из файла или генерирует их некоторым образом).
   2. Процесс 2: выполняет сложение
   3. Процесс 3: выводит результат
2. Выполнить задачу, разбив её на 3 потока.
   1. Поток 1: заполняет данными входные матрицы (читает из файла или генерирует их некоторым образом).
   2. Поток 2: выполняет сложение
   3. Поток 3: выводит результат
3. Разбить сложение на P потоков.

Исследовать зависимость между количеством потоков, размерами входных данных и параметрами целевой вычислительной системы.

## Выполнение работы.

**Сложение матриц с помощью 3 процессов.**

Для создания новых процессов используется функция fork(). Процесс-ребенок выполняет всю логику, а родительский процесс ждет его выполнения с помощью функции wait().

Для коммуникации между процессами используется разделяемая память. Для создания разделяемой памяти использовалась функция mmap().

**Сложение матриц с помощью 3 потоков.**

Создание потоков происходит с помощью std::future. Было решено использовать std::future вместо std::thread, т.к. std::future позволяет получить значение, возвращаемое из функции, которая запускается в новом потоке.

**Сложение матриц с помощью P потоков.**

Для сложения матриц с помощью P=4 потоков использовался вектор потоков, которые суммируют “углы” матриц.

**Исследование зависимости между количеством потоком, размерами входных данных и параметрами вычислительной системы.**

В таблице 1 представлено сравнение размера входных данных и времени вычисления при выполнении в одном потоке:

Таблица 1 - Сравнение размера входных данных и времени вычисления

| Время вычисления(милисек.) | Размер входных данных |
| --- | --- |
| 17 | 1000 x 1000 |
| 445 | 5000 x 5000 |
| 1524 | 10000 x 10000 |
| 4111 | 15000 x 15000 |

В таблице 2 представлено сравнение размера входных данных и времени вычисления при распределении на 4 потоков.

Таблица 2 - Сравнение размера входных данных и времени вычисления

| Время вычисления(милисек.) | Размер входных данных |
| --- | --- |
| 33 | 1000 x 1000 |
| 590 | 5000 x 5000 |
| 2662 | 10000 x 10000 |
| 7173 | 15000 x 15000 |

## Выводы.

В ходе выполнения лабораторной работы была реализована программа на языке программировании С++, получены навыки работы с потоками и процессами, а так же было установлено, что разбиение вычислительных задач на несколько потоков, положительно сказывается на скорости выполнения программы.