



**Спецкурс: системы и средства параллельного  
программирования**

**Отчёт № 3**

**Параллельный алгоритм поиска простых чисел в  
заданном диапазоне с помощью "решета Эратосфена"**

Работу выполнила  
**Зайденварг Елизавета**

Москва 2020

# Постановка задачи и формат данных

## Задача

Реализовать параллельный алгоритм поиска простых чисел в заданном диапазоне с помощью "решета Эратосфена".

Результатом работы является количество простых чисел, выведенное в стандартный поток вывода, и файл с самими числами в текстовом виде (сортировать не обязательно).

Использовать MPI (в первой части) и pthread (во второй части).

Оценить суммарное время выполнения для всех процессов и максимальное время выполнения среди всех процессов в зависимости от числа процессов. Во время выполнения не включать время ввода/вывода.

Тестирование проводить на Polus.

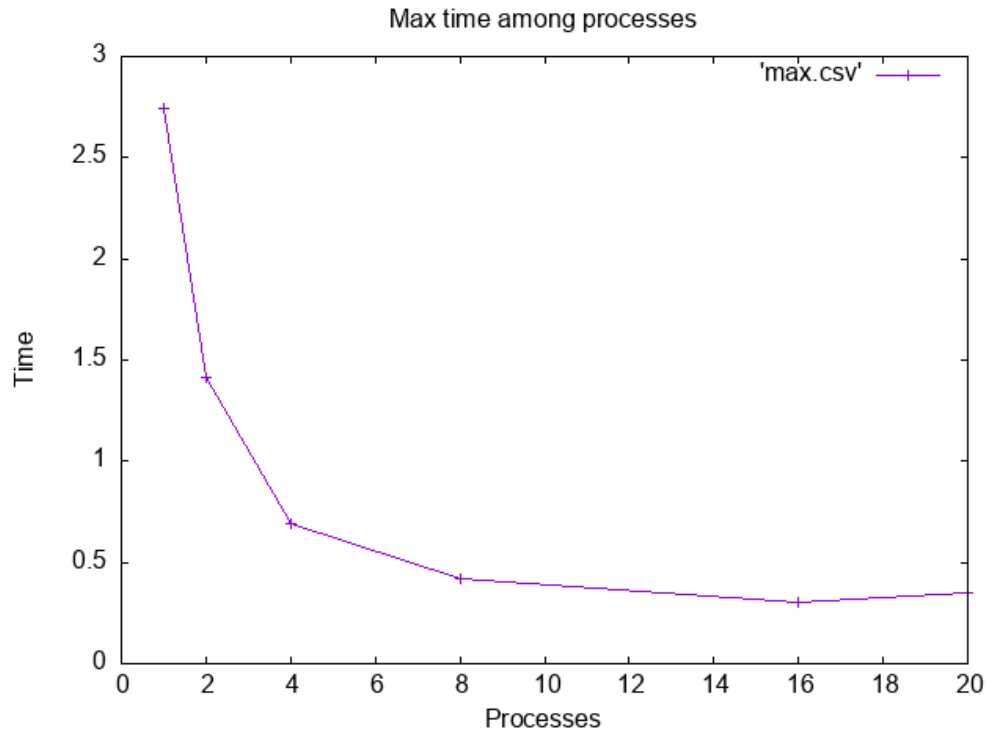
## Формат командной строки

<первое число из диапазона> <последнее число из диапазона> <имя выходного файла для хранения списка простых чисел в текстовом виде через пробелы>

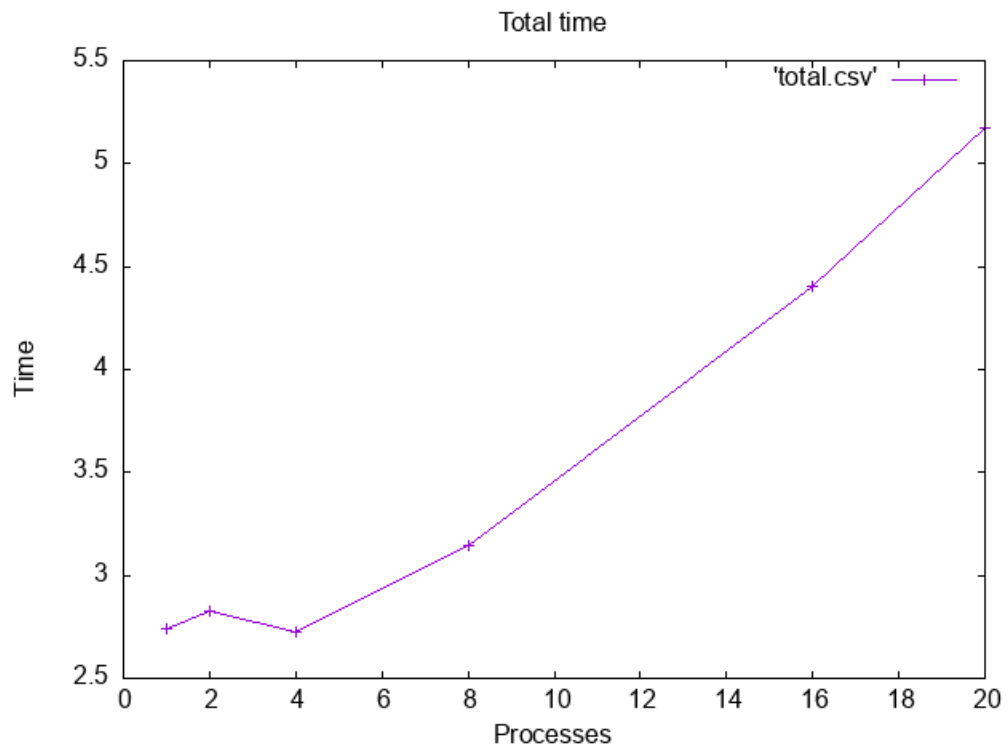
# Результаты

Приведены результаты для диапазона  $[1, 10^8]$ .

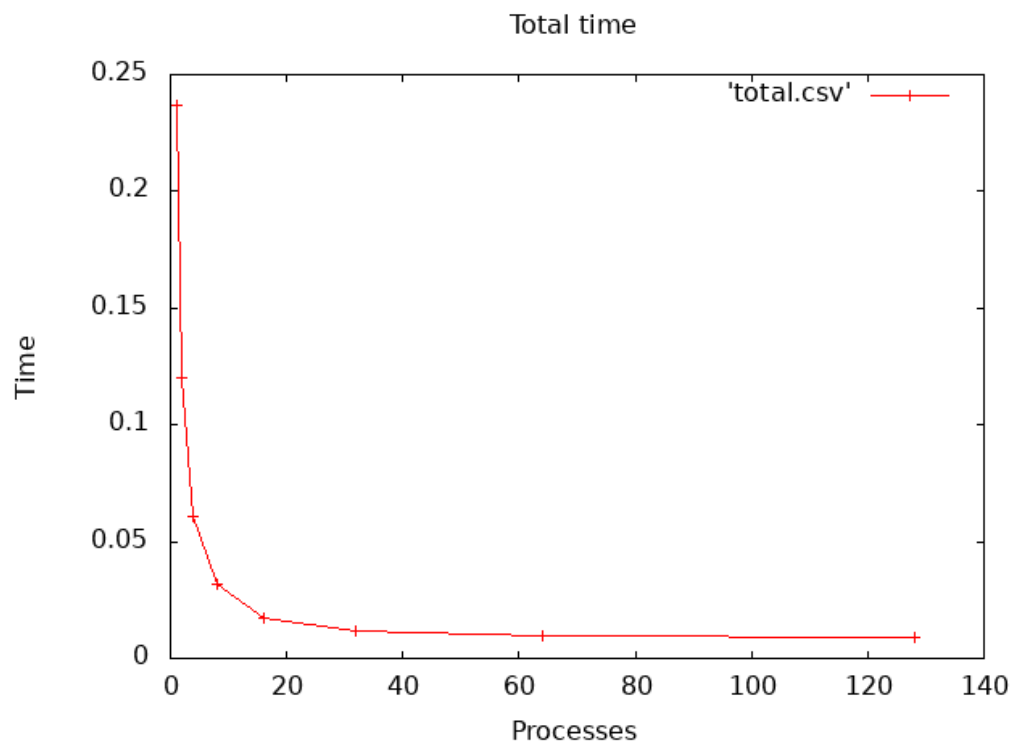
## 1. Максимальное время выполнения среди всех процессов в зависимости от числа процессов



## 2. Суммарное время выполнения всех процессов в зависимости от числа процессов



### 3. pthread – суммарное время выполнения всех тредов



## Выводы

Исследования показали следующие результаты:

1. Наименьшее время выполнения среди всех процессов достигается при 16 процессах.
2. Наименьшее суммарное время выполнения всех процессов достигается при 4 процессах.
3. Наименьшее время выполнения pthreads достигается на 128 тредах
4. Минимальное время на pthreads значительно меньше времени на MPI. Это обоснованно тем, что накладные расходы на создание тредов намного меньше накладных расходов на создание процесса, и тем, что для тредов нет необходимости разделять память.