

Московский Государственный Университет им. М.В. Ломоносова  
Факультет Вычислительной Математики и Кибернетики  
Кафедра Суперкомпьютеров и Квантовой Информатики



## **Спецкурс: системы и средства параллельного программирования.**

### **Отчёт № 3.**

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

Работу выполнил  
**Тони Кастильо Мартин**

Москва 2018

## Постановка задачи и формат данных.

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.

**Задача:** Результатом работы является количество простых чисел выведенное в командную строку и файл с самими числами в текстовом виде (сортировать не обязательно).

**Формат командной строки:** <число  $A$ > <число  $B$ > (диапазон чисел)

**Формат файла-матрицы:** Матрица представляются в виде бинарного файла следующего формата:

| Тип               | Значение                | Описание                    |
|-------------------|-------------------------|-----------------------------|
| Число типа size_t | $A$ – натуральное число | Первый элемент диапазона    |
| Число типа size_t | $B$ – натуральное число | Последнее элемент диапазона |

## Описание алгоритма.

**Математическая постановка:** Алгоритм “Решета Эратосфена” можно представить в следующем алгоритме:

**Algoritmo** Criba de Eratóstenes (Complejidad  $O(n \log \log n)$ )

**Entrada:** Un número natural  $n$

**Salida:** El conjunto de números primos anteriores a  $n$  (incluyendo  $n$ )

1. Escriba todos los números naturales desde 2 hasta  $n$

2. **Para**  $i$  desde 2 hasta  $\lfloor \sqrt{n} \rfloor$  **haga lo siguiente:**

1. **Si**  $i$  no ha sido marcado **entonces:**

1. **Para**  $j$  desde  $i$  hasta  $n \div i$  **haga lo siguiente:**

1. Ponga una marca en  $i \times j$

3. **El resultado es:** Todos los números sin marca

**Анализ времени выполнения:** Для оценки времени выполнения программы использовалась функция:

- MPI\_Wtime

Для повышения надёжности экспериментов опыты проводились несколько раз (10).

**Верификация:** Для проверки корректности работы программы использовались тестовые данные.

**Основные функции:**

- **Разбор командной строки.** В рамках функции осуществляется анализ и разбор командной строки.

## Результаты выполнения

Зависимость времени выполнения рабочих циклов: для диапазона чисел:  $[1, 10^8]$  для 2, 4, 8, 16 и 32 процессоров.

