ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ  
 НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
 «ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет компьютерный наук

Департамент программной инженерии

Домашнее задание к 4 по курсу Архитектура вычислительных систем.

Разработка многопоточных приложений с использованием OpenMP

Вариант 14

Пояснительная записка

Исполнитель:

Студент группы БПИ-196

Кузнецов Михаил Федорович

Москва, 2020

# Постановка задачи

Определить множество индексов i, для которых A[i] и B[i] не имеют общих делителей (единицу в роли делителя не рассматривать). Входные данные: массивы целых положительных чисел А и B, произвольной длины ≥ 1000. Количество потоков является входным параметром.

# Выбор модели приложения

Была выбрана модель «Итеративный параллелизм». Так как в программе используется реализация нескольких идентичных потоков, каждый из которых содержит циклы. Потоки программы, описываются итеративными функциями и работают совместно над решением одной задачи.

# Описание алгоритма

Для установки количества потоков, выделения области распараллеливания и создание переменной для каждого потока, для подсчета количества обработанных потоком элементов используется #pragma omp parallel num\_threads(nThread) firstprivate(count). Для определения какие индексы какой поток обрабатывает используеться #pragma omp for schedule(static, bl). Размер блока, который обрабатывает поток, равен числу элементов разделить на число потоков, но не меньше 500, чтоб потоки не часто менялись. При этом число элементов равно длине наименьшего массива. Далее каждый поток обрабатывает свою часть массивов. Для каждой пары элементов сравнивая их НОД (Наибольший общий делитель) с единицей, так как если их НОД единица, то у них нет общих делителей больше единицы. Индексы всех подходящий элементов поток записывает в вектор с использованием #pragma omp critical (push).

# Формат входных данных

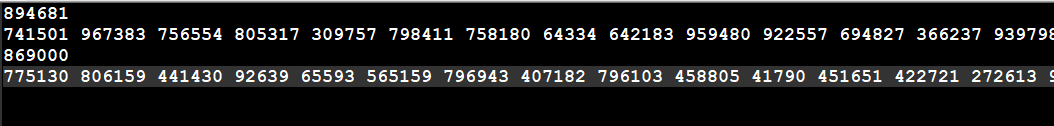
## Формат вызова из командной строки

Программы запускается из командной строки с тремя аргументами: первый имя входного файла, второй имя выходного файла, а третий количество потоков.



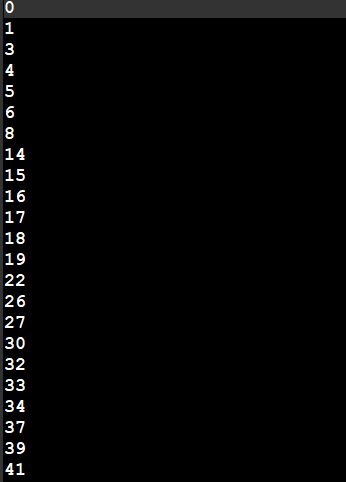
## Формат входного файла

На первой строке указывается размер массива A. На второй строке находиться массив А. На третей строке находиться размер массива В. На четвертой сам массив B.



# Формат выходных данных

В выходном файле располагаются индексы подходящих элементов, каждый индекс на новой строке.

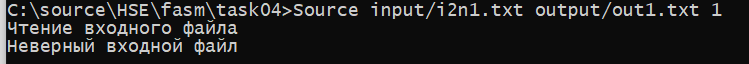


# Тестирование программы

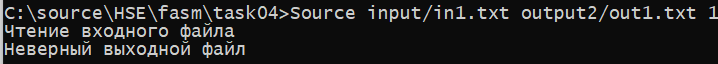
Было приготовлено 5 рандомно сгенерированных тестовых файлов для проверки, они находятся в папке input. А в папке output ответы к этим тестам.

## Тесты некорректного ввода

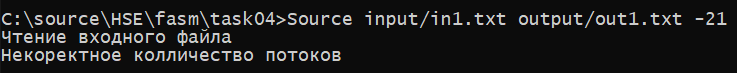
### Некорректный входной файл



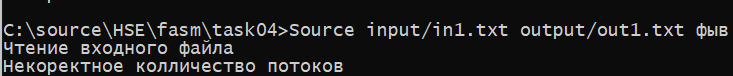
### Некорректный выходной файл



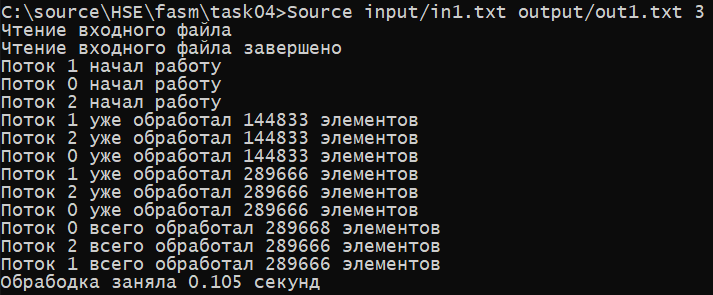
### Отрицательное число потоков

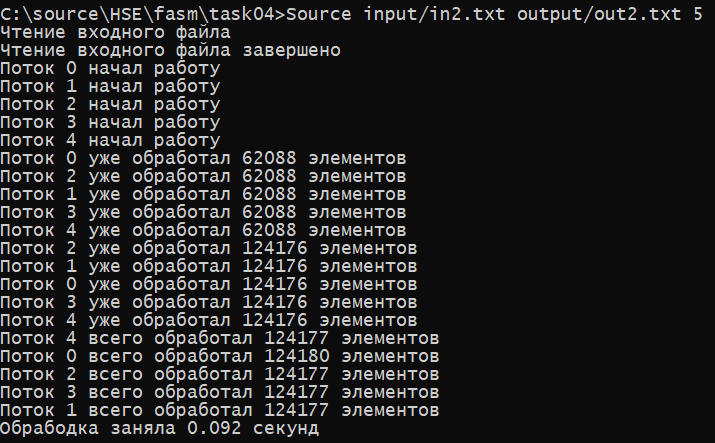


### Неверный формат потока

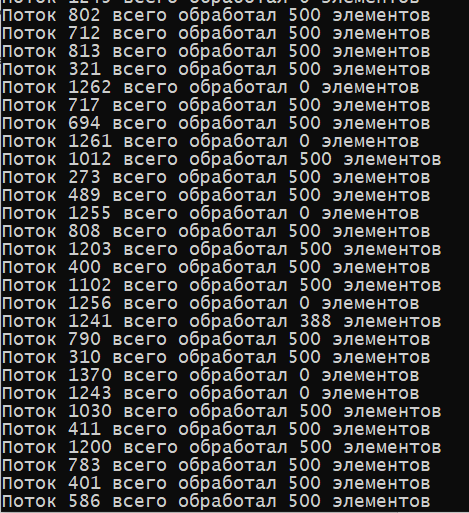


## Тесты корректной работы





### Тест с большим количеством потоков



# Источники

Пункт 4 из предложенного списка литературы по многопоточности

Грегори Р. Эндрюс. Основы многопоточного, параллельного и распределенного программирования. - М.: Издательский дом "Вильямс 2003.