

## Параллельная сортировка слиянием

Дан массив  $a$  из  $n$  целых чисел в случайном порядке. Нужно упорядочить массив в порядке возрастания.

Алгоритм состоит из трех шагов:

- Разбить массив  $a$  на куски (чанки).
- Отсортировать каждый чанк массива (любым алгоритмом сортировки). Чанки нужно сортировать параллельно (относительно друг друга).
- Слить все чанки в один упорядоченный массив, используя параллельный алгоритм слияния.

## Задание

Реализовать параллельную сортировку слиянием с помощью OpenMP.

Программа должна принимать на вход целочисленные  $n$ ,  $m$ ,  $P$ , где  $m$  – максимальный размер чанка,  $P$  – число OpenMP потоков.

В файле `stats.txt` должно быть записано время работы сортировки и перечислены входные параметры.

В первой строчке файла `data.txt` должен быть записан исходный массив, во второй строчке файла `data.txt` – отсортированный массив.

Пример:

```
$ ./run 8 4 1
$ cat stats.txt
0.00001s 8 4 1
$ cat data.txt
8 6 7 2 1 3 4 5 9
1 2 3 4 5 6 7 8 9
```

Произвести запуск при  $P = 1, 2, 4, 8, 16$ .

Составить график зависимости  $T(P)$ ,  $S(P)$ ,  $E(P)$  при фиксированном значении  $n$  и  $m$  ( $m$  нужно подобрать оптимальным для вашей вычислительной системы). На графике  $T(P)$  отобразить время работы `qsort` из `stdlib.h`.

## Примечания

Нужно использовать директиву `pragma omp tasks` и(или) `pragma omp sections`.

*Минимальное условие для успешной сдачи решения:* в многопоточном режиме на объёмах данных ( $>10$ мб) решение должно работать быстрее `qsort` из `stdlib.h`.