

Задание 4

Параллельная сортировка слиянием

Отчёт

Кучеров В.Д.

2022

1. Постановка задачи

Реализовать параллельный алгоритм сортировки слиянием с использованием программного интерфейса POSIX Threads. Массив для сортировки считывается из входного файла. Массив разбивается на подмассивы (чанки), каждый чанк сортируется последовательно, слияние чанков происходит параллельно. На вход программа должна принимать имя входного файла-массива, имя выходного файла-массива, натуральное число p , где p — число нитей (POSIX Threads), проводящих непосредственно сортировку чанков. Составить график зависимости $T(p)$ (время), $S(p)$ (ускорение) при фиксированном входном массиве размера $n = 100\,000\,000$.

2. Реализация

В ходе работы были реализованы три программы: `thread_merge_sort.cpp`, `is_sorted.cpp`, `array_generator.cpp`, отвечающие соответственно за сортировку, проверку отсортированности массива, генерацию массива. Был выбран рекурсивный алгоритм сортировки:

1. Разбиение массива на чанки
2. Запускается сначала один поток, которому передаются все чанки
3. Если в функцию передан один чанк, то производится сортировка стандартными средствами языка и возвращается отсортированный чанк.
4. Иначе, полученные чанки разбиваются на две группы, каждая из которых сортируется в новом потоке. После происходит слияние двух полученных отсортированных массивов.

3. Формат командной строки

```
./thread_merge_sort <входной_файл_массив> <выходной_файл_массив> <число_нитей>  
./is_sorted <входной_файл_массив>  
./array_generator <количество_элементов> <выходной_файл_массив> <ядро_генерации>
```

4. Спецификация системы

Процессор: Intel(R) Core(TM) i9-9880H CPU @ 2.30GHz

Число вычислительных ядер: 8

5. Результаты выполнения

Для каждого числа нитей проводилось 3 эксперимента (3 различных массива длины 100 000 000 элементов), в таблице представлено время каждого эксперимента, усреднённое время и ускорение.

| Число нитей n | Эксперимент 1 (с) | Эксперимент 2 (с) | Эксперимент 3 (с) | Среднее время работы (с) | Ускорение |
|-----------------|-------------------|-------------------|-------------------|--------------------------|-----------|
| 1 | 7,814097 | 8,094139 | 7,868201 | 7,925 | 1,000 |
| 2 | 4,556397 | 4,676118 | 4,651764 | 4,628 | 1,712 |
| 3 | 3,776356 | 3,765084 | 3,795756 | 3,779 | 2,097 |
| 4 | 3,121532 | 3,152072 | 3,122516 | 3,132 | 2,530 |
| 5 | 3,024371 | 3,021835 | 3,013528 | 3,020 | 2,624 |
| 6 | 2,725288 | 2,698225 | 2,766201 | 2,730 | 2,903 |
| 7 | 2,544844 | 2,549244 | 2,529562 | 2,541 | 3,119 |
| 8 | 2,410415 | 2,405319 | 2,393323 | 2,403 | 3,298 |



