МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

“Московский государственный университет геодезии и картографии”

(МИИГАИК)

Факультет геоинформатики и информационной безопасности

Кафедра геоинформационных систем и технологий

**Лабораторная работа №4**

**"Знакомство с параллелизацией"**

Проверил: Выполнил:

Лебедев Евгений Денисович Студент группы: 2024-ФГИИБ-ПИ-1б

Центнер Валерий Викторович

Москва 2025

**Вариант 28**

Формулировка задания:

~ Реализовать параллельную версию быстрой сортировки с

использованием C++.

~ Провести замеры времени выполнения для последовательной и

параллельной версий алгоритма для массивов разного размера.

~ Рассчитать коэффициенты ускорения и представить результаты в

виде таблиц и графиков.

~ Сравнить результаты с лабораторной работой №3.

~ Сделать выводы об эффективности параллелизации корректности

выбора размера массива N.

Ссылка на GitHub репозиторий с файлами:

<https://github.com/guguker/InfAlgoLebedev/tree/main/second_semestr>

Замер данных на моём ноутбуке:

БС – быстрая сортировка

БС\_П – быстрая параллельная сортировка (и число потоков)

Все данные даны в секундах.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Размер массива | БС | БС\_П 2 | БС\_П 4 | БС\_П 8 |
| 1000 | 0 | 0 | 0 | 0 |
| 10000 | 0.001 | 0 | 0 | 0 |
| 100000 | 0.014 | 0.009 | 0.006 | 0.005 |
| 1000000 | 0.172 | 0.152 | 0.124 | 0.069 |
| 10000000 | 1.8 | 1.7 | 0.867 | 0.956 |
| 100000000 | 28.3 | 15 | 17 | 8.7 |
| 1000000000 | 1209.1 | 876.5 | 579 | 621 |

~20 мин. ~14 мин. ~10 мин. ~10 мин.

Сравнение получившегося графика и графика из ЛР №3:

Получившийся график (с) :  
Изображение выглядит как снимок экрана, График, текст

Содержимое, созданное искусственным интеллектом, может быть неверным.

График из ЛР №3 (с) :  
Изображение выглядит как снимок экрана, диаграмма, линия, График

Содержимое, созданное искусственным интеллектом, может быть неверным.

Расчёт коэффициента ускорения:

Вычисляем Speedup по данной формуле:

Изображение выглядит как текст, Шрифт, снимок экрана, линия

Содержимое, созданное искусственным интеллектом, может быть неверным.

Таблица с КФ ускорения (в секундах):

Speedup = SD (и число потоков)

БС – быстрая сортировка

БС\_П – быстрая параллельная сортировка (и число потоков)

Все данные даны в секундах.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Размер массива | БС | БС\_П\_2 | БС\_П\_4 | БС\_П\_8 | SD\_2 | SD\_4 | SD\_8 |
| 1000 | 0 | 0 | 0 | 0 | - | - | - |
| 10000 | 0.001 | 0 | 0 | 0 | - | - | - |
| 100000 | 0.014 | 0.009 | 0.006 | 0.005 | 1.56 | 2.33 | 2.8 |
| 1000000 | 0.172 | 0.152 | 0.124 | 0.069 | 1.13 | 1.39 | 2.49 |
| 10000000 | 1.8 | 1.7 | 0.867 | 0.956 | 1.06 | 2.08 | 1.88 |
| 100000000 | 28.3 | 15 | 17 | 8.7 | 1.89 | 1.66 | 3.25 |
| 1000000000 | 1209.1 | 876.5 | 579 | 621 | 1.38 | 2.09 | 1.95 |

~20 мин. ~14 мин. ~10 мин. ~10 мин.

Как отработало ускорение?

Ускорение отработало как должно, в среднем коэффициент Speedup был равен 1.97, что на мой взгляд является отличным способом ускорить происходящие процессы.  
Максимальный эффект ускорение дало на 100 млн. при 8 потоков.

Выводы о подборе N и пояснение о используемом методе:

Можно сделать вывод, что чем больше массив, тем выгоднее использовать многопоточность.

При малых массивах (до 100 000) многопоточность может вообще не давать выигрыша, а иногда — даже медленнее из-за затрат на создание и синхронизацию потоков.

На массивах от 1 млн и выше многопоточность становится эффективной.

На миллиардных объёмах — почти обязательно, если важна производительность.

Что такое mutex?

std::mutex — это механизм, который гарантирует, что только один поток может получить доступ к определённому участку кода в один момент времени.