

84. ИСПОЛЬЗОВАНИЕ АЛГОРИТМОВ ШЕЛЛА И ХОАРА ДЛЯ РЕШЕНИЯ ЭКОНОМИЧЕСКИХ ЗАДАЧ

Рябычина А. С., студент гр.577602

*Белорусский государственный университет информатики и радиоэлектроники¹
г. Минск, Республика Беларусь*

Кириенко Н.А. – кандидат технических наук, доцент

Аннотация. В работе рассматривается применение алгоритмов сортировки Шелла и Хоара для решения экономических задач обработки больших данных. Проанализированы сценарии использования сортировок в экономике. Описана теоретическая разница между простыми и усовершенствованными методами. Проведен эксперимент на массивах. Результаты подтверждают целесообразность метода сортировки Хоара для экономического анализа больших данных.

Сортировка массивов является одним из основополагающих инструментов в экономическом анализе и обработке и организации больших данных [1]. В финансовом секторе алгоритмы сортировки упорядочивают миллионы транзакций по времени, сумме или риску. В оптимизации инвестиционных портфелей сортировка активов по доходности/волатильности позволяет быстро находить оптимальные комбинации. В ритейле сортировка заказов по приоритету (например, цена, популярность, локация) оптимизирует логистику. Сортировка также применяется в макроэкономическом моделировании для оценки и ранжирования эффективности отдельных единиц в разные периоды [2]. Сортировки обеспечивают высокую скорость и точность экономического анализа. В банковских системах сортировка транзакций ускоряет поиск аномалий, а в логистике помогает оптимизировать маршруты поставок. Простые сортировки (методы пузырька, вставки, выбора) неэффективны для больших данных из-за квадратичной зависимости от размера массива. Из-за этого они эффективны для только массивов небольших размеров. Усовершенствованные алгоритмы решают эту проблему. Сортировка Шелла (улучшенный метод вставки) сначала работает с элементами, находящимися на большом расстоянии друг от друга, далее уменьшает расстояние, что позволяет быстрее перемещать элементы на свои места. Сортировка Хоара (быстрая, QuickSort) является рекурсивным методом, основанным на принципе «разделяй и властвуй». Данная сортировка предполагает выбор опорного элемента, деление массива на группы: меньшие, равные и большие, чем опорные. Каждая из этих групп сортируется рекурсивно.

Целью эксперимента является сравнение эффективности сортировок Шелла [3] и Хоара [4] на массивах случайных целых чисел от -1000 до 1000. Тестируемые размеры массивов: от 10 000 до 1 000 000 элементов с шагом 10,000. Для каждого размера проводится 10 запусков, заполняя массивы случайно, копируя в буферы для каждой сортировки. Время измеряется через функцию clock(). Средние значения сохраняются в структуру tests_res. На экран выводятся результаты сортировок каждого размера (по 10 раз на размер), а также средние значения в формате для преобразования в таблицу.

Основные шаги разработанного алгоритма следующие.

1. Установить $n=10000$.
2. Если $n \leq 1000000$, перейти к шагу 4, иначе завершить.
3. Выделить динамическую память под три массива размером n элементов. Инициализировать.
4. Установить счетчик $i=1$.
5. Если $i < 11$, перейти к шагу 6, иначе к шагу 9.
6. Сгенерировать случайный массив. Скопировать его в два вспомогательных массива. Замерить время выполнения сортировки Шелла и быстрой сортировки. Добавить измеренные времена к значению среднего. Вывести результат. Увеличить i на 1 и перейти к шагу 5.
7. Вычислить средние времена работы программы. Сохранить результаты в массив структур для текущего n .
8. Освободить память. Увеличить n на 10000 и перейти к шагу 2.
9. Вывести заголовок "СРЕДНЕЕ" и средние времена для всех n из массива структур.

Проанализируем графики, построенные на средних значениях времени выполнения сортировок методами Шелла и Хоара по 10 запускам. Экспериментальные данные показывают время выполнения в секундах (среднее по 10 запускам). Исследуя график, можно заявлять о том, что метод Хоара стабильно быстрее метода Шелла для всех размеров массива, но особенно при сортировке 100 тыс.-1 млн элементов, благодаря рекурсивному разделению. График представлен в системе зависимости времени от размера массива: график скорости работы алгоритма, сортирующего методом Хоара, ниже графика скорости

работы алгоритма, сортирующего методом Шелла, но оба растут как логарифмические функции (рисунок 1).

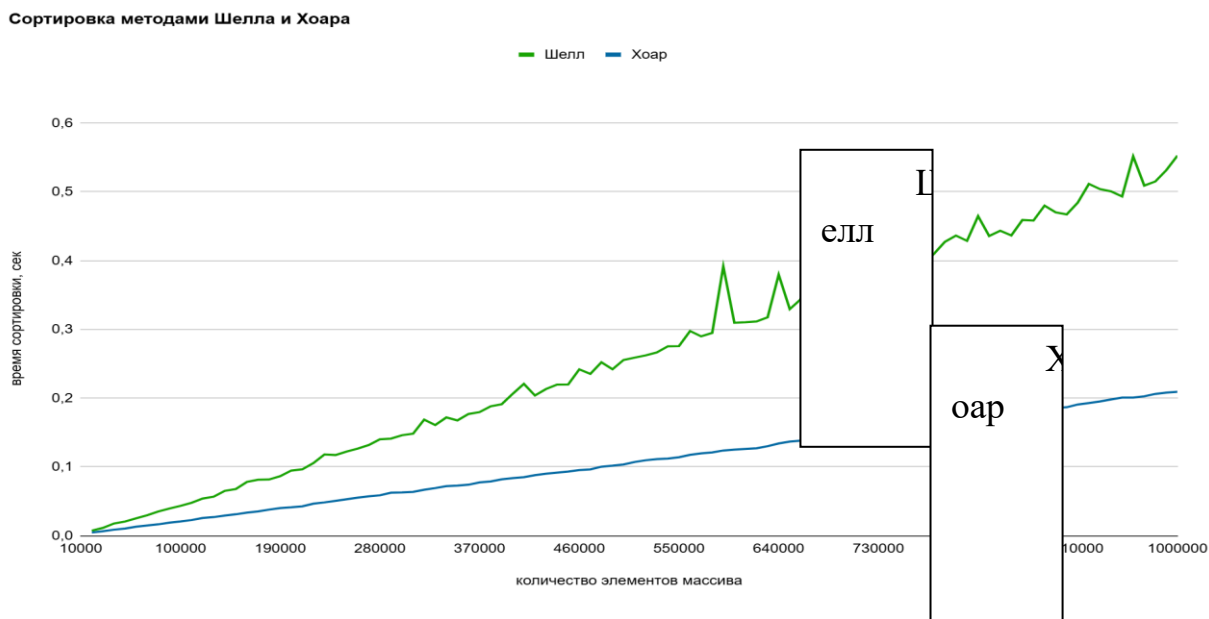


Рисунок 1 – Графики скорости работы алгоритмов

Рассчитаем, насколько метод Хоара эффективнее метода Шелла. Назовем это значение ускорением. Рассчитаем его как $(t_{\text{Shell}} - t_{\text{Hoar}}) / t_{\text{Shell}} \times 100\%$, где t – среднее время в секундах. Среднее ускорение по всем размерам составляет 57.5%, медианное – 58.6%, для малых ($n \leq 105$) – 47.3%, для больших ($n > 5 \cdot 10^5$) – 60.4%. Сравнение показало превосходство метода Хоара над методом Шелла по времени для случайных массивов больших размеров, что идеально для экономических задач с большими данными (разница составила 36-62% (10^4 - 10^6 элементов)). Оба алгоритма масштабируемы, но метод Хоара выигрывает за счет оптимального разделения массива на группы.

Заключение

Проведенные исследования подтвердили превосходство сортировки Хоара над методом Шелла для обработки больших массивов экономических данных — среднее ускорение 57,5%, на больших объемах до 60,4%. Оба алгоритма эффективны для финансового анализа транзакций, оптимизации портфелей и логистики, но QuickSort оптимален для реального времени. Полученные результаты рекомендуют применять сортировку Хоара в системах экономического моделирования больших данных.

Список использованных источников:

- [1] Теплов А.В. Применение алгоритмов сортировки и поиска при обработке экономических данных [Электронный ресурс] // Казанский государственный энергетический университет. – Режим доступа: <https://birskin.ru/index.php/2012-02-07-11-31-02/57-iii-/1434-2025-11-17-06-26-43?format=pdf.ijeais>
- [2] Сортировщики: Основы, применение и будущее алгоритмов сортировки [Электронный ресурс] // VseKilledzhi.ru. – Режим доступа: <https://vsekilledzhi.ru/articles/sortirovschiki-osnovy-primenenie-i-buduschee-algoritmov-sortirovki/theknowledgeacademy>
- [3] Горков А. Эффективные алгоритмы сортировки (BoAT_03_effective_sort) [Электронный ресурс] // Agorkov.ru. – Режим доступа: https://agorkov.ru/wp-content/uploads/papers/BoAT_03_effective_sort.pdf.ftc
- [4] Описание алгоритмов сортировки и сравнение их производительности [Электронный ресурс] // Habr.com. – Режим доступа: <https://habr.com/ru/articles/335920/ijsret>