Отчет по лабораторной работе №1

М4О-205Б-23

Кудаков Т.Г.

Вариант 5 (Insert + Quick)

1. **Описание реализованных алгоритмов сортировки:**
   1. *Сортировка вставками (Insertion Sort)*

Проходит по массиву слева направо. Каждый новый элемент вставляет в отсортированную часть массива, перемещая большие элементы вправо.

* 1. *Быстрая сортировка (Quick Sort)*

Выбираем опорный элемент (pivot). Разделяем массив: элементы меньше pivot идут влево, больше — вправо. Рекурсивно повторяем для левой и правой частей.

* 1. *std::sort*

Использует Introsort: сначала Quick Sort, при глубокой рекурсии переключается на Heap Sort, а для маленьких массивов − Insertion Sort.

1. **График времени исполнения по трем сортировкам для лучшего случая:**



Рисунок 1 − График лучшего случая для трех сортировок

1. **График времени исполнения по трем сортировкам для худшего случая:**

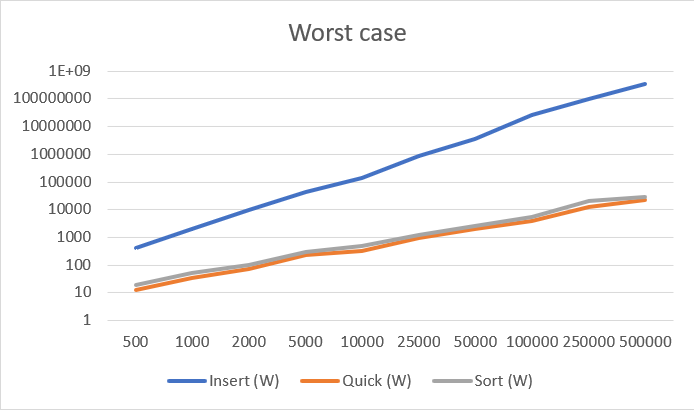


Рисунок 2 − График худшего случая для трех сортировок

1. **График среднего времени исполнения по трем сортировкам:**

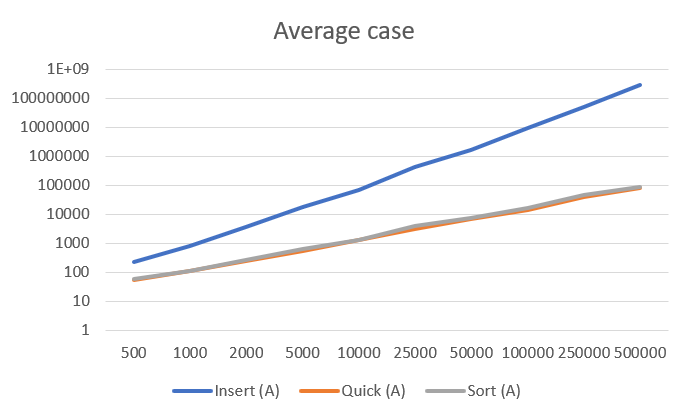


Рисунок 3 − График среднего времени для трех сортировок

1. **Вывод по лабораторной работе:**

Разные алгоритмы сортировки подходят для разных задач. Сортировка вставками проста и эффективна для небольших или почти отсортированных массивов. Быстрая сортировка работает быстро в среднем случае, но может быть медленной при неудачном выборе опорного элемента. std::sort в C++ − оптимизированное решение, использующее гибридный подход для высокой производительности в любых условиях.

1. **Используемая литература:**
   1. Quick Sort in C++ (with Code ) - Scaler Topics // scaler Topics URL: <https://www.scaler.com/topics/quick-sort-in-cpp/>
   2. Форматирование строк и ввода-вывода (современный C++) // Microsoft Learn URL: <https://learn.microsoft.com/ru-ru/cpp/text/string-and-i-o-formatting-modern-cpp?view=msvc-170>