**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра РАПС**

**отчет**

**по лабораторным работам № 6-7**

**по дисциплине «Программирование и основы алгоритмизации»**

**Тема:** Алгоритмы сортировки, алгоритмы поиска

|  |  |  |
| --- | --- | --- |
| Студент гр. 3401 |  | Орлов Д.Р. |
| Преподаватель |  | Армашев А.А. |

Санкт-Петербург

2024

**Цели работ:**

Сравнительный анализ различных методов сортировки массивов.

Сравнительный анализ различных методов поиска данных в массивах.

## Ход работы.

**Сортировка неупорядоченного массива.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Сортировка/кол-во элементов | Прямым включением | Прямым выбором | Прямым обменом | Шейкерная | Быстрая |
| 25 000 | 0.141 | 0.156 | 0.203 | 0.172 | 0.000 |
| 50 000 | 0.640 | 0.578 | 0.828 | 0.719 | 0.000 |
| 75 000 | 1.531 | 1.250 | 1.734 | 2.109 | 0.000 |
| 100 000 | 2.672 | 2.141 | 3.125 | 4.704 | 0.015 |

Таблица 1.

Исходя из данных Таблицы 1, можно сделать вывод, что самой эффективной сортировкой массива является *Быстрая сортировка*.

**Сортировка массива размерностью 100 000.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Сортировка/ тип массива | Прямым включением | Прямым выбором | Прямым обменом | Шейкерная | Быстрая |
| Неупорядочен-ный | 2.758  2506270673250520854 | 2.098  4999992951  99992 | 3.232 4999950000  229976273 | 4.647  3333048424  25017524 | 0.015 2225343  585482 |
| Прямой порядок | 0.000  99999  0 | 2.204  49999500000 | 2.109  4999950000  0 | 0.000 99999  0 | 0.016  1602677  0 |
| Обратный порядок | 6.141  49999498414999945011 | 4.953 4999938325  55274 | 5.932 4999962298  4999934072 | 5.327 4999949999  4999428846 | 0.000 1612303  144256 |

Таблица 2.

Исходя из данных Таблицы 2, можно сделать вывод, что *быстрая сортировка* является самой быстрой для сортировки произвольного массива из-за самого маленького числа сравнений элементов. Для упорядоченного массива сортировка прямым включением имеет самое малое число сравнений, исходя из этого можно сделать вывод что для частично упорядоченного массива она будет самой быстрой. Для обратно упорядоченного массива самой быстрой является быстрая сортировка.

**Поиск элемента в массива.**

|  |  |  |  |
| --- | --- | --- | --- |
| Поиск/кол-во элементов | Линейный | Двоичный | Интерполяционный |
| 25 000 | 5107 | 13 | 3 |
| 50 000 | 10630 | 15 | 5 |
| 75 000 | 37497 | 16 | 5 |
| 100 000 | 59585 | 17 | 5 |

Таблица 3.

По данным таблицы 3, можно сделать вывод, что при выборе алгоритма поиска стоит учитывать структуру данных и размер массива. Интерполяционный и двоичный поиск оказываются предпочтительными для больших массивов, в то время как линейный подходит только для небольших структур или для случаев, когда данные не отсортированы.

## Вывод.

Для работы с упорядоченными длинными массивами данных оптимальными методами являются быстрая сортировка, двоичный поиск и интерполяционный поиск. Для коротких неупорядоченных или частично упорядоченных массивов подойдут линейный поиск и сортировка вставками.

Производительность зависит от условий, которые следует учитывать во время написания программы.