**Практическое задание 8**

**Тема.** Эффективные алгоритмы сортировки.

**Цель.** Получить навыки по анализу вычислительной сложности нескольких алгоритмов сортировки и определение наиболее эффективного алгоритма.

**Задание.**

Разработать три алгоритма сортировки, определенные вариантом. Провести анализ вычислительной и емкостной сложности алгоритма на массивах, заполненных случайно. Определить наиболее эффективный алгоритм.

1. Разработать алгоритм простой сортировки, определенной вариантом. Определить емкостную и временную сложность алгоритма.
2. Разработать алгоритм усовершенствованной сортировки, определенной вариантом. Определить емкостную и временную сложность алгоритма.
3. Разработать алгоритм сортировки методом простого слияния. Определить емкостную и временную сложность алгоритма.
4. Провести контрольные прогоны функций на уже отсортированных массивах, отсортированных в обратном порядке и сгенерированных с помощью генератора псевдослучайных чисел различных размеров (N>999). Рабочие прогоны функций должны проводиться на одинаковых массивах. Во время сортировки вычислять время её выполнения t. Провести эмпирическую (практическую) оценку вычислительной сложности алгоритмов для трех случаев, для чего предусмотреть в программе подсчет фактического количества операций сравнения С и количества операций перемещения М. Полученные результаты свести в сводные таблицы.
5. Представить графики зависимости С+М от N и t от N для анализируемых алгоритмов в трех случаях.
6. Провести анализ зависимости алгоритмов сортировок от размера и исходной упорядоченности массива по составленным таблицам. Определить эффективный алгоритм для каждого случая.

Таблица 1. Пример сводной таблицы результатов тестирования программы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N** | **t** | **Tа(n)=N5** | **Tэ=С+M** | **Tэ/Tа** |
| 60000 | 12.2536 | 3600000000 | 3009960000 | 0,8361 |
| 70000 | 14.3364 | 4900000000 | 4145400000 | 0,846 |
| 80000 | 16.8701 | 6400000000 | 5313920000 | 0,8303 |
| 900000 | 23.7769 | 8100000000 | 6840450000 | 0,8445 |
| 100000 | 24.9365 | 10000000000 | 8493000000 | 0,8493 |

Таблица 2. Варианты заданий

|  |  |  |  |
| --- | --- | --- | --- |
| **Вариант** | **Алгоритм простой сортировки** | **Алгоритм усовершенствованной сортировки** | **Алгоритм**  **слияния** |
| 1 | Простого обмена (пузырек) | Шейкерная | Простое слияние |
| 2 | Простого обмена (пузырек) с условием Айверсона | Хоара | Простое слияние |
| 3 | Простого обмена (пузырек) с условием Айверсона | Шейкерная с условием Айверсона | Простое слияние |
| 4 | Простой вставки | Шелла со смещениями Д. Шелла | Простое слияние |
| 5 | Простой вставки | Шелла со смещениями Д. Кнута | Простое слияние |
| 6 | Простой вставки | Шелла со смещениями Р. Седжвика | Простое слияние |
| 7 | Простого выбора | Пирамидальная сортировка (heap) | Простое слияние |
| 8 | Простого выбора | Турнирная сортировка | Простое слияние |

**Примечания.**

Методы определения смещения для сортировки Шелла.

Перед выполнением сортировки происходит вычисление длин промежутков d, которые записываются в массив.

Метод Шелла.

di=N/2i; N, N/2, …, 2, 1.

Метод Кнута.

di=(3i-1)/2; 1, 4, 13, 40, 121, …. (не более N/3)

Метод Седжвика.

di=9\*(2i-2i/2)+1; (i – четное)

di=8\*2i-6\*2(i+1)/2+1; (i – нечетное)

Дополнительно, разумеется, берем d0=1: 1, 5, 19, 41, 109, 209, 505.