Министерство образования и науки РФ

Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский Нижегородский государственный университет

им. Н.И. Лобачевского»

Институт информационных технологий, математики и механики

**Отчет по учебной практике**

**МЕТОДЫ СОРТИРОВКИ**

Выполнил: Клюев Дмитрий Андреевич, студент группы 381806-1

Проверил: к. т. н., доцент кафедры МОСТ Кустикова

Нижний Новгород

2018

Содержание

[Введение 3](#_Toc533377887)

[1. Постановка задачи 4](#_Toc533377888)

[2. Руководство пользователя 5](#_Toc533377889)

[3. Руководство программиста 6](#_Toc533377890)

[Структура программы 6](#_Toc533377891)

[Описание алгоритмов 6](#_Toc533377892)

[Сортировка выбором 6](#_Toc533377893)

[Сортировка вставками 6](#_Toc533377894)

[Сортировка пузырьком 7](#_Toc533377895)

[Сортировка подсчетом 8](#_Toc533377896)

[Быстрая сортировка 10](#_Toc533377897)

[Сортировка слиянием 11](#_Toc533377898)

[Описание функций 12](#_Toc533377899)

[4. Заключение 15](#_Toc533377900)

[Анализ методов сортировки 15](#_Toc533377901)

[Литература 16](#_Toc533377902)

Введение

1. Постановка задачи

Разработать прототип файлового менеджера с функцией показа файлов в заданном каталоге, упорядоченных по возрастанию размера. Для этого необходимо реализовать следующие методы сортировки:

* Сортировка выбором
* Сортировка вставками
* Сортировка пузырьком
* Сортировка подсчетом
* Быстрая сортировка
* Сортировка слиянием

Входные данные:

* Путь до директории, в которой необходимо отсортировать содержимое
* Метод сортировки

Выходные данные:

* Отсортированный список имен файлов
* Размер файлов
* Время сортировки

1. Руководство пользователя

После запуска выводится следующее сообщение [Рис. 1]. Выбор режима осуществляется вводом соответствующего номера.

C:\Folder\1.PNG

1. Выбор режима

Выбор направления сортировки [Рис. 2].

C:\Folder\2.PNG

1. Выбор порядка сортировки

Ввод пути к директории. [Рис. 3].

C:\Folder\0.PNG

1. Ввод адреса

При некорректном вводе будут выведены следующие сообщения [Рис. 4], [Рис. 5].

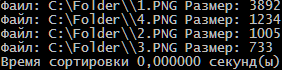
C:\Folder\3.PNG

1. Сообщение об ошибке

C:\Folder\4.PNG

1. Сообщение об ошибке

При корректном вводе будут выведены результаты [Рис. 6].



1. Список файлов, время сортировки
2. Руководство программиста

Структура программы

Программа состоит из одного файла *main.c,* который содержит функцию main(), функции сортировок, функцию вывода отсортированных названий файлов, их размеров и времени сортировки.

Описание алгоритмов

Сортировка выбором

1. Находим позицию минимального элемента в массиве
2. Меняем минимальный элемент с первым
3. Продолжаем для неотсортированной части массива Таблица 1
4. Сортировка вставками

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Индекс | *0* | *1* | *2* | *3* | *4* | *5* | *6* | *7* | Индекс минимального элемента | Текущий элемент |
| Итерация | Массив | | | | | | | |
| 1 | 1 | 0 | 2 | -4 | 9 | 6 | 5 | 3 | 3 | 0 |
| 2 | **-4** | 0 | 2 | 1 | 9 | 6 | 5 | 3 | 1 | 1 |
| 3 | **-4** | **0** | 2 | 1 | 9 | 6 | 5 | 3 | 3 | 2 |
| 4 | **-4** | **0** | **1** | 2 | 9 | 6 | 5 | 3 | 3 | 3 |
| 5 | **-4** | **0** | **1** | **2** | 9 | 6 | 5 | 3 | 7 | 4 |
| 6 | **-4** | **0** | **1** | **1** | **3** | 6 | 5 | 9 | 6 | 5 |
| 7 | **-4** | **0** | **1** | **1** | **3** | **5** | 6 | 9 | 6 | 6 |
| 8 | **-4** | **0** | **1** | **1** | **3** | **5** | **6** | 9 | 7 | 7 |
| – | **-4** | **0** | **1** | **1** | 1. 3 | **5** | **6** | **9** | – | – |

Сортировка вставками

1. Начнаем с одноэлементного массива (он уже отсортирован)
2. Увеличивем рассматриваемый массив на 1
3. Первый элемент из неотсортированной части продвигаем влево, пока левая часть не станет отсортированной [Таблица 3]
4. Сортровка вставками

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Итерация | Массив | | | | | | | | Индекс сравниваемого элемента | Нужно ли менять с предыдущим |
| *0* | *1* | *2* | *3* | *4* | *5* | *6* | *7* |
| 1 | **5** | 9 | 8 | 2 | 6 | 3 | 1 | 4 | 1 | - |
| 2 | **5** | **9** | 8 | 2 | 6 | 3 | 1 | 4 | 2 | + |
| **5** | 8 | **9** | 2 | 6 | 3 | 1 | 4 | 1 | - |
| 3 | **5** | **8** | **9** | 2 | 6 | 3 | 1 | 4 | 3 | + |
| **5** | **8** | 2 | **9** | 6 | 3 | 1 | 4 | 2 | + |
| **5** | 2 | **8** | **9** | 6 | 3 | 1 | 4 | 1 | + |
| 2 | **5** | **8** | **9** | 6 | 3 | 1 | 4 | 0 | - |
| 4 | **2** | **5** | **8** | **9** | 6 | 3 | 1 | 4 | 4 | + |
| **2** | **5** | **8** | 6 | **9** | 3 | 1 | 4 | 3 | + |
| **2** | **5** | 6 | **8** | **9** | 3 | 1 | 4 | 2 | - |
| 5 | **2** | **5** | **6** | **8** | **9** | 3 | 1 | 4 | 5 | + |
| **2** | **5** | **6** | **8** | 3 | **9** | 1 | 4 | 4 | + |
| **2** | **5** | **6** | 3 | **8** | **9** | 1 | 4 | 3 | + |
| **2** | **5** | 3 | **6** | **8** | **9** | 1 | 4 | 2 | + |
| **2** | 3 | **5** | **6** | **8** | **9** | 1 | 4 | 1 | - |
| 6 | **2** | **3** | **5** | **6** | **8** | **9** | 1 | 4 | 6 | + |
| **2** | **3** | **5** | **6** | **8** | 1 | **9** | 4 | 5 | + |
| **2** | **3** | **5** | **6** | 1 | **8** | **9** | 4 | 4 | + |
| **2** | **3** | **5** | 1 | **6** | **8** | **9** | 4 | 3 | + |
| **2** | **3** | 1 | **5** | **6** | **8** | **9** | 4 | 2 | + |
| **2** | 1 | **3** | **5** | **6** | **8** | **9** | 4 | 1 | + |
| 1 | **2** | **3** | **5** | **6** | **8** | **9** | 4 | 0 | - |
| 7 | **1** | **2** | **3** | **5** | **6** | **8** | **9** | 4 | 7 | + |
| **1** | **2** | **3** | **5** | **6** | **8** | 4 | **9** | 6 | + |
| **1** | **2** | **3** | **5** | **6** | 4 | **8** | **9** | 5 | + |
| **1** | **2** | **3** | **5** | 4 | **6** | **8** | **9** | 4 | + |
| **1** | **2** | **3** | 4 | **5** | **6** | **8** | **9** | 3 | - |
| – | **1** | **2** | **3** | **4** | **5** | **6** | **8** | **9** | – | – |

Сортировка пузырьком

1. Проходим массив с начала до конца
2. Если элемент с большим номером имеет меньшее значение, делаем замену
3. Сдвигаем конец влево [Таблица 4]

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Итерация | Массив | | | | | | | | Индексы сравниваемых элементов | Нужно ли менять элементы местами |
| *0* | *1* | *2* | *3* | *4* | *5* | *6* | *7* |
| 1 | 8 | 6 | 4 | 2 | 9 | 5 | 3 | 7 | 0, 1 | + |
| 6 | 8 | 4 | 2 | 9 | 5 | 3 | 7 | 1, 2 | + |
| 6 | 4 | 8 | 2 | 9 | 5 | 3 | 7 | 2, 3 | + |
| 6 | 4 | 2 | 8 | 9 | 5 | 3 | 7 | 3, 4 | - |
| 6 | 4 | 2 | 8 | 9 | 5 | 3 | 7 | 4, 5 | + |
| 6 | 4 | 2 | 8 | 5 | 9 | 3 | 7 | 5, 6 | + |
| 6 | 4 | 2 | 8 | 5 | 3 | 9 | 7 | 6, 7 | + |
| 2 | 6 | 4 | 2 | 8 | 5 | 3 | 7 | **9** | 0, 1 | + |
| 4 | 6 | 2 | 8 | 5 | 3 | 7 | **9** | 1, 2 | + |
| 4 | 2 | 6 | 8 | 5 | 3 | 7 | **9** | 2, 3 | - |
| 4 | 2 | 6 | 8 | 5 | 3 | 7 | **9** | 3, 4 | + |
| 4 | 2 | 6 | 5 | 8 | 3 | 7 | **9** | 4, 5 | + |
| 4 | 2 | 6 | 5 | 3 | 8 | 7 | **9** | 5, 6 | + |
| 3 | 4 | 2 | 6 | 5 | 3 | 7 | **8** | **9** | 0, 1 | + |
| 2 | 4 | 6 | 5 | 3 | 7 | **8** | **9** | 1, 2 | - |
| 2 | 4 | 6 | 5 | 3 | 7 | **8** | **9** | 2, 3 | + |
| 2 | 4 | 5 | 6 | 3 | 7 | **8** | **9** | 3, 4 | + |
| 2 | 4 | 5 | 3 | 6 | 7 | **8** | **9** | 4, 5 | - |
| 4 | 2 | 4 | 5 | 3 | 6 | **7** | **8** | **9** | 0, 1 | - |
| 2 | 4 | 5 | 3 | 6 | **7** | **8** | **9** | 1, 2 | - |
| 2 | 4 | 5 | 3 | 6 | **7** | **8** | **9** | 2, 3 | + |
| 2 | 4 | 3 | 5 | 6 | **7** | **8** | **9** | 3, 4 | - |
| 5 | 2 | 4 | 3 | 5 | **6** | **7** | **8** | **9** | 0, 1 | - |
| 2 | 4 | 3 | 5 | **6** | **7** | **8** | **9** | 1, 2 | + |
| 2 | 3 | 4 | 5 | **6** | **7** | **8** | **9** | 2, 3 | - |
| 6 | 2 | 3 | 4 | **5** | **6** | **7** | **8** | **9** | 0, 1 | - |
| 2 | 3 | 4 | **5** | **6** | **7** | **8** | **9** | 1, 2 | - |
| 7 | 2 | 3 | **4** | **5** | **6** | **7** | **8** | **9** | 0, 1 | - |
| – | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | – | – |

Сортировка подсчетом

1. Создаём массив количества вхождений n-ного элемента; длина равна разности наибольшего и наименьшего элемента
2. Проходя по исходному массиву, прибавляем 1 к номеру со значением текущего элемента
3. Заполняем исходный массив с начала; количество включений данного элемента равно значению во вспомогательном массиве [Таблица 5]
4. Сортировка подсчётом

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *min = 0*  *max = 3* | Массив | | | | | | | | Дополнительный массив | | | |
| *0* | *1* | *2* | *3* | *4* | *5* | *6* | *7* | *0* | *1* | *2* | *3* |
| Итерации | 3 | 2 | 3 | 0 | 1 | 1 | 3 | 3 | 0 | 0 | 0 | 0 |
| 1 | ↑ |  |  |  |  |  |  |  | 0 | 0 | 0 | 1 |
|  | ↑ |  |  |  |  |  |  | 0 | 0 | 1 | 1 |
|  |  | ↑ |  |  |  |  |  | 0 | 0 | 1 | 2 |
|  |  |  | ↑ |  |  |  |  | 1 | 0 | 1 | 2 |
|  |  |  |  | ↑ |  |  |  | 1 | 1 | 1 | 2 |
|  |  |  |  |  | ↑ |  |  | 1 | 2 | 1 | 2 |
|  |  |  |  |  |  | ↑ |  | 1 | 2 | 1 | 3 |
|  |  |  |  |  |  |  | ↑ | 1 | 2 | 1 | 4 |
| 2 | 0 | 2 | 3 | 0 | 1 | 1 | 3 | 3 | ↑ |  |  |  |
| **0** | 1 | 1 | 0 | 1 | 1 | 3 | 3 |  | ↑ |  |  |
| **0** | **1** | **1** | 2 | 1 | 1 | 3 | 3 |  |  | ↑ |  |
| **0** | **1** | **1** | **2** | 3 | 3 | 3 | 3 |  |  |  | ↑ |
| – | **0** | **1** | **1** | **2** | **3** | **3** | **3** | **3** | – | | | |

Быстрая сортировка

1. Выбираем ведущий элемент (из середины или случайно)
2. Все меньшие элементы перемещаем в левую часть, большие – в правую
3. Рекурсивно сортируем левую и правую части, пока они не станут одноэлементными [Таблица 6]
4. Быстрая сортировка

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Индекс | *0* | *1* | *2* | *3* | *4* | *5* | *6* | *7* |
| Шаг | Массив | | | | | | | |
| 1 | 3 | 7 | 4 | 6 | 4 | 5 | 8 | 7 |
| 3 | 5 | 4 | 6 | 4 | 7 | 8 | 7 |
| 2 | 3 | 5 | 4 | 4 | **6** | 7 | 8 | 7 |
| 3 | 4 | 4 | **5** | 7 | 7 | **8** |
| 3 | 3 | 4 | 4 | 7 | 7 |
| - | **3** | **4** | **4** | **5** | **6** | **7** | **7** | **8** |

Сортировка слиянием

1. Рекурсивно разбиваем массив на половины, пока они не станут одноэлементными
2. Упорядоченно сливаем упорядоченные подмассивы [Таблица 7]

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Индекс | *0* | *1* | *2* | *3* | *4* | *5* | *6* | *7* |
| Шаг | Массив | | | | | | | |
| 1 | 2 | 8 | 4 | 1 | 0 | 2 | 8 | 3 |
| 2 | 8 | 4 | 1 | 0 | 2 | 8 | 3 |
| 2 | 8 | 1 | 4 | 0 | 2 | 8 | 3 |
| 2 | 8 | 1 | 4 | 0 | 2 | 8 | 3 |
| 2 | 2 | 8 | 1 | 4 | 0 | 2 | 3 | 8 |
| 1 | 2 | 4 | 8 | 0 | 2 | 3 | 8 |
| 3 | 2 | 2 | 4 | 8 | 0 | 2 | 3 | 8 |
| – | **0** | **1** | **2** | **2** | **3** | **4** | **8** | **8** |

Описание функций

void Print(wchar\_t \*\*name, ULONGLONG \*size, int \*index, int n, int order, double time)

**Назначение:** Печать списка файлов, размеров, времени сортировки.

**Входные параметры:**

\*\*name – массив строк имён файлов

\*size – массив размеров

\*index – массив соответствий номеров элементам

n – количество файлов

order – порядок сортировки (возрастание, убывание)

time – время, затраченное на сортировку

**Выходные данные:** Нет

void ChooseSort(ULONGLONG \*size\_copy, int n)

**Назначение:** сортировка выбором

**Входные параметры:**

\*size\_copy – копия массива размеров

n – количество файлов для сортировки

**Выходные данные:** Нет

void InsertSort(ULONGLONG \*size\_copy, int n)

**Назначение:** сортировка вставками

**Входные параметры:**

\*size\_copy – копия массива размеров

n – количество файлов для сортировки

**Выходные данные:** Нет

void BubbleSort(ULONGLONG \*size\_copy, int n)

**Назначение:** сортировка пузырьком

**Входные параметры:**

\*size\_copy – копия массива размеров

n – количество файлов для сортировки

**Выходные данные:** Нет

void CountingSort(ULONGLONG \*size\_copy, int n)

**Назначение:** сортировка подсчётом

**Входные параметры:**

\*size\_copy – копия массива размеров

n – количество файлов для сортировки

**Выходные данные:** Нет

void QuickSort(ULONGLONG \*size\_copy, int n1, int n2)

**Назначение:** быстрая сортировка

**Входные параметры:**

\*size\_copy – копия массива размеров

n1 – начало интервала сортировки

n2 – конец интервала сортировки

**Выходные данные:** Нет

void QuickSplit(ULONGLONG \*size\_copy, int \*i, int \*j, ULONGLONG base)

**Назначение:** разбиение массива на большую и меньшую часть

**Входные параметры:**

\*size\_copy – копия массива размеров

\*i, \*j – индексы левой и правой границы разбиваемого подмассива

base – значение ведущего элемента

**Выходные данные:** Нет

void MergeSort(ULONGLONG \*size\_copy, int l, int r)

**Назначение:** сортировка слиянием

**Входные параметры:**

\*size\_copy – копия массива размеров

n1 – начало интервала сортировки

n2 – конец интервала сортировки

**Выходные данные:** Нет

void Merge(ULONGLONG \*size\_copy, int l, int m, int r)

**Назначение:** упорядоченное слияние упорядоченных массивов

**Входные параметры:**

\*size\_copy – копия массива размеров

l левая граница сливаемых массивов

m – граница сливаемых массивов

r – правая граница сливаемых массивов

**Выходные данные:** Нет

int ListDirectoryContents(wchar\_t \*sDir, wchar\_t \*\*name, ULONGLONG \*size, int mode, int \*amount)

**Назначение:** Получение файлов и их размеров в директории

**Входные параметры:**

\*sDir – введённый адрес

\*\*name – массив имён файлов в директории

\*size – массив размеров файлов

mode – режим сортировки

\*amount – количество файлов в директории

**Выходные данные:**

Существование введённой директории

1. Заключение

В ходе выполнения практической работы был разработан прототип файлового менеджера с функцией показа файлов в заданном каталоге, упорядоченных по возрастанию или убыванию размера с помощью следующих методов сортировки:

* Сортировка выбором
* Сортировка вставками
* Сортировка пузырьком
* Сортировка подсчетом
* Быстрая сортировка
* Сортировка слиянием

Анализ методов сортировки

1. Анализ методов сортировки

|  |  |  |
| --- | --- | --- |
| Метод сортировки | Время выполнения | Дополнительная память |
| Сортировка выбором | O(n2) | O(1) |
| Сортировка вставками | O(n2) | O(1) |
| Пузырьковая сортировка | O(n2) | O(1) |
| Сортировка подсчетом | O(n) | max(n) – min(n) + 1 |
| Быстрая сортировка | O(n·log2 n) | O(1) |
| Сортировка слиянием | O(n·log2 n) | O(n) |

В [Таблица 8] показана зависимость времени выполнения сортировки от количества сортируемых элементов

Литература