**Министерство науки и высшего образования Российской Федерации** ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**Факультет безопасности информационных технологий Дисциплина:**

«Алгоритмы и структуры данных»

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 1**

«Сортировка числового массива с использованием алгоритма Shakersort»

**Выполнил:**

Чу Ван Доан N3247



(подпись)

**Проверил:**

Ерофеев С. А.



(подпись)

Санкт-Петербург 2024 г.

# СОДЕРЖАНИЕ

[Введение 4](#_heading=h.1fob9te)

1. [Описание функционала программы 5](#_heading=h.2et92p0)
2. [Блок-схема 6](#_heading=h.3dy6vkm)
3. [Код программы 7](#_heading=h.4d34og8)
4. [Результаты тестирования 8](#_heading=h.17dp8vu)

[Заключение 10](#_heading=h.26in1rg)

# ВВЕДЕНИЕ

Задача работы – Разработать программу шейкерной сортировки массива чисел из файла.

Шейкерная сортировка (Shaker Sort) является алгоритмом сортировки, который использует подход, аналогичный пузырьковой сортировке.

Для реализации был выбран язык программирования C++.

# ОПИСАНИЕ ФУНКЦИОНАЛА ПРОГРАММЫ

Описание алгоритма Shakersort:

* Открывается файл с именем "data.txt" для чтения, и его содержимое читается и сохраняется в вектор dataArray с помощью функции readFileContent;
* Вызывается функция shakerSort, которая сортирует содержимое вектора dataArray с использованием алгоритма шейкерной сортировки;
* Открывается файл с именем "data.txt" для записи, и отсортированный массив сохраняется в этот файл с помощью функции writeArrayToFile;
* Открывается файл с именем "data.txt" для чтения, и содержимое файла выводится на экран с помощью функции printFileContent, чтобы показать отсортированный массив.
* Учитывается диапазон массива от -1,7 \* 10^308 до 1,7 \* 10^308.

Алгоритм шейкерной сортировки, реализованный в функции shakerSort, имеет следующие шаги:

1. Инициализируются переменные left и right, указывающие на левую и правую границы сегмента массива, который нужно отсортировать. Изначально left равно 0, а right равно последнему индексу массива.
2. Устанавливается флаг swapped в значение true, чтобы начать первую итерацию.
3. Пока swapped равен true, выполняем следующие действия:

* Устанавливается swapped в значение false;
* Проход от левого индекса left до правого индекса right – 1;
* Если текущий элемент больше следующего элемента, выполняется обмен и устанавливается swapped в значение true;
* Если swapped остается false, цикл прерывается.

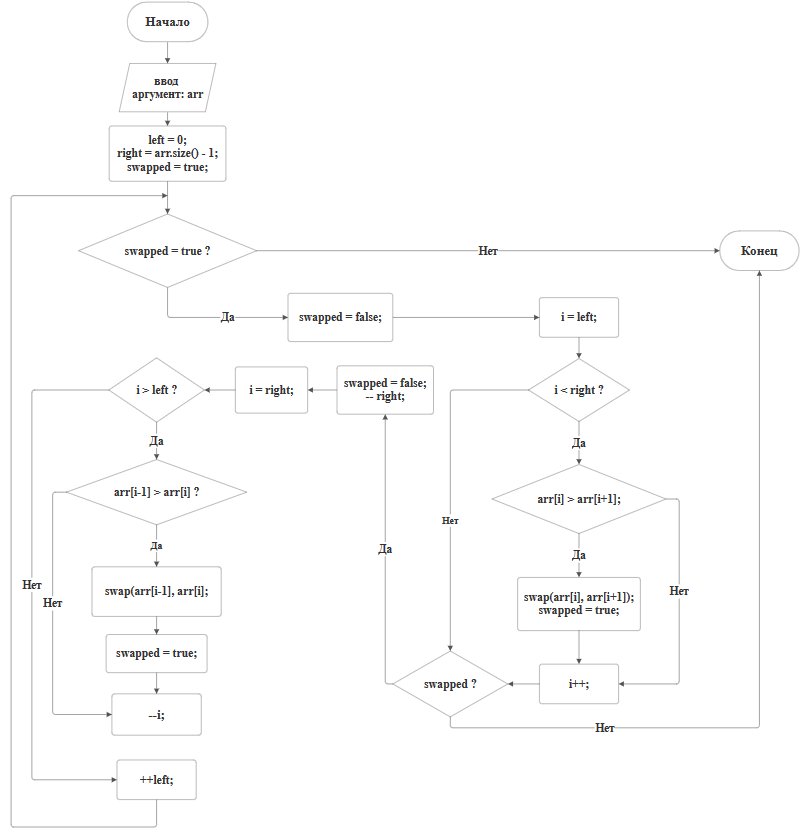
1. Если swapped равно false, цикл прерывается.
2. Уменьшается значение right на 1.
3. Пока swapped равно true, выполняются следующие действия:

* Устанавливается swapped в значение false;
* Проход от правого индекса right до левого индекса left + 1;
* Если текущий элемент меньше предыдущего элемента, выполняется обмен и устанавливается swapped в значение true;
* Если swapped остается false, цикл прерывается.

1. Увеличивается значение left на 1.
2. Повторяются шаги 3-7 до тех пор, пока left не превысит right.

Таким образом, после завершения работы функции shakerSort, массив arr будет отсортирован по возрастанию.

# БЛОК-СХЕМА

****

**Рисунок 1 – Блок-схема алгоритма Shakersort**

# КОД ПРОГРАММЫ

#include <iostream>

#include <fstream>

#include <cstdlib>

#include <ctime>

#include <vector>

using namespace std;

void printFileContent(const char \*filename);

void readFileContent(const char \*filename, vector<int> &arr);

void shakerSort(vector<int> &arr);

void writeArrayToFile(const char \*filename, const vector<int> &arr);

int main()

{

const char \*filename = "data.txt";

vector<int> dataArray;

readFileContent(filename, dataArray);

cout << "Array before sorting: " << endl;

printFileContent(filename);

cout << endl;

shakerSort(dataArray);

writeArrayToFile(filename, dataArray);

cout << "Array after sorting:" << endl;

printFileContent(filename);

return 0;

}

// Функция для чтения и печати всего содержимого файла

void printFileContent(const char \*filename)

{

ifstream file(filename);

if (!file.is\_open())

{

cout << "The file cannot be opened for reading." << endl;

return;

}

int number;

while (file >> number)

{

cout << number << " ";

}

file.close();

}

// Функция для чтения содержимого из файла и сохранения в массив

void readFileContent(const char \*filename, vector<int> &arr)

{

ifstream file(filename);

if (!file.is\_open())

{

cout << "The file cannot be opened for reading." << endl;

return;

}

// Считать данные из файла и сохранить в массив

int value;

while (file >> value)

{

arr.push\_back(value);

}

file.close();

}

// Функция для сортировки массива с использованием алгоритма Shaker Sort

void shakerSort(vector<int> &arr)

{

int left = 0;

int right = arr.size() - 1;

bool swapped = true;

while (swapped)

{

swapped = false;

// Проход от левого к правому

for (int i = left; i < right; ++i)

{

if (arr[i] > arr[i + 1])

{

swap(arr[i], arr[i + 1]);

swapped = true;

}

}

if (!swapped)

{

break;

}

swapped = false;

--right;

// Проход от правого к левому

for (int i = right; i > left; --i)

{

if (arr[i - 1] > arr[i])

{

swap(arr[i - 1], arr[i]);

swapped = true;

}

}

++left;

}

}

// Функция для записи отсортированного массива в файл

void writeArrayToFile(const char \*filename, const vector<int> &arr)

{

ofstream file(filename);

if (!file.is\_open())

{

cout << "The file cannot be opened for writing." << endl;

return;

}

for (const auto &value : arr)

{

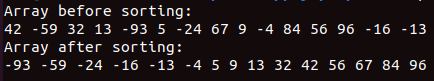
file << value << " ";

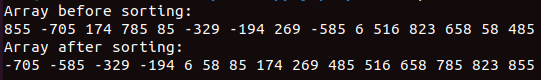
}

file.close();

}

# РЕЗУЛЬТАТЫ ТЕСТИРОВАНИЯ

****

****

# ЗАКЛЮЧЕНИЕ

В ходе выполнения данной лабораторной работы, я разработала и реализовала алгоритм сортировки числового массива с использованием алгоритма Shakersort (шейкерная сортировка).

Алгоритм Shakersort является вариацией пузырьковой сортировки, при которой проходы по массиву выполняются в обе стороны, сначала слева направо, а затем справа налево. Это позволяет эффективнее перемещать наибольшие и наименьшие элементы к краям массива.

Программа начинается с чтения содержимого файла "data.txt" и сохранения числовых данных в вектор dataArray с помощью функции readFileContent. Затем вызывается функция shakerSort, которая сортирует массив вектора dataArray по возрастанию с использованием алгоритма Shakersort.

После сортировки отсортированный массив записывается обратно в файл "data.txt" с помощью функции writeArrayToFile. Затем происходит вывод отсортированного массива на консоль с помощью функции printFileContent.

В результате выполнения программы в файле "data.txt" будет содержаться отсортированный массив, а на консоль будет выведен отсортированный массив для проверки результатов.

Таким образом, лабораторная работа успешно реализует сортировку числового массива с использованием алгоритма Shakersort на языке программирования C++.