**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра САПР**

**ОТЧЁТ**

**По лабораторной работе №4**

**По дисциплине «Программирование»**

Студент гр. 3351 \_\_\_\_\_\_\_\_\_\_\_\_ Морозов А.А.

Преподаватель \_\_\_\_\_\_\_\_\_\_\_\_ Рыжов Н.Г.

Санкт-Петербург

2023

цель работы

Организовать начальное заполнение массива с помощью функции генерации случайного (псевдослучайного) числа rand(). Дан массив целых чисел. Написать программу сортировки массива по возрастанию методом «пузырька», используя функции srand() и rand() для заполнения массива. Размерность массива - 1000. Придумать и реализовать собственный алгоритм сортировки. Сравнить алгоритмы, подсчитав (в программе) количество используемых операций сравнения в каждом из них при одних и тех же исходных данных. Представить в отчете вывод об их эффективности.

**Текст программы с сортировкой “пузырёк”**

#include <iostream>

#include <cstdlib>

#include <ctime>

#include <conio.h>

using namespace std;

int compCount = 0; // Переменная для подсчёта количества операций сравнения

void SortMoves(int arr[], int n) {

bool replace;

for (int i = 0; i < n - 1; i++) {

replace = false;

for (int j = 0; j < n - i - 1; j++) {

compCount++; // Подсчёт количества операций сравнения

if (arr[j] > arr[j + 1]) { // Обмен элементов

int temp = arr[j];

arr[j] = arr[j + 1];

arr[j + 1] = temp;

replace = true;

}

}

if (!replace) { // Если в этом проходе не было обменов, значит массив уже отсортирован

break;

}

}

}

int main() {

setlocale(LC\_ALL, "RU");

srand(static\_cast<unsigned int>(time(0))); // Инициализация генератора случайных чисел с помощью текущего времени

const int size = 100; // Размер массива

const int maxrand = 200; // Выбор максимальной границы для генерации псевдослучайных чисел

int sum = 0; // Переменная для суммы элементов массива

double average, sized = size;

int array[size]; // Создание и заполнение массива случайными числами

for (int i = 0; i < size; i++) {

array[i] = rand()%maxrand;

}

SortMoves(array, size); // Вызов функции сортировки пузырьком

cout << "Сгенерированный массив: "; // Вывод массива

for (int i = 0; i < size; i++) {

cout << "\n" << "Число: " << array[i] << "\t"; // Вывод чисел построчно

sum += array[i]; // Подсчёт суммы

cout << "Сумма: " << sum; // Вывод суммы построчно

}

average = sum / sized; // Поиск среднего значения

cout << "\n" << "Среднее значение: " << average; // Вывод среднего значения

cout << endl;

cout << "Количество операций сравнения: " << compCount;

\_getch();

return 0;

}

**Текст программы с сортировкой “выборка”**

#include <iostream>

#include <cstdlib>

#include <ctime>

#include <conio.h>

using namespace std;

int compCount = 0; // Переменная для подсчёта количества операций сравнения

int option(int list[], int start, int length) { // Функция сортировки выборкой

int min = start;

for (int i = start; i < length; i++) {

compCount++; // Подсчёт количества операций сравнения

if (list[i] < list[min])

min = i;

}

return min;

}

int main() {

setlocale(LC\_ALL, "RU");

const int size = 100; // Размер массива

const int maxrand = 200; // Выбор максимальной границы для генерации псевдослучайных чисел

int sum = 0; // Переменная для суммы элементов массива

bool Sorted = true;

double average, sized = size;

int array[size]; // Создание и заполнение массива случайными числами

srand(static\_cast<unsigned int>(time(0))); // Генерация случайных чисел с помощью текущего времени

for (int i = 0; i < size; i++) {

array[i] = rand() % maxrand;

}

for (int i = 0; i < size - 1; i++) { // Вызов функции сортировки

int minIndex = option(array, i, size);

int temp = array[i];

array[i] = array[minIndex];

array[minIndex] = temp;

}

cout << "Сгенерированный массив: "; // Вывод массива

for (int i = 0; i < size; i++) {

cout << "\n" << "Число: " << array[i] << "\t"; // Вывод чисел построчно

sum += array[i]; // Подсчёт суммы

cout << "Сумма: " << sum; // Вывод суммы построчно

}

average = sum / sized; // Поиск среднего значения

cout << "\n" << "Среднее значение: " << average; // Вывод среднего значения

cout << endl;

cout << "Количество операций сравнения: " << compCount;

\_getch();

return 0;

}

**Описание программы**

Программное обеспечение – Microsoft Visual Studio 2022. Язык программирования – C++. При запуске программа создаёт массив из случайных (псевдослучайных) целых чисел, далее сортирует его по возрастанию одним из методов (пузырёк или выборка). Потом в консоль выводится отсортированный массив (рядом с каждым числом выводится сумма всех предыдущих построчно), среднее значение всех чисел в массиве и количество операций сравнения в ходе сортировки.

**Руководство пользователя**

При запуске программа сразу выдаёт результат по заранее заготовленным программистом переменным. В консоль выводится сгенерированный и отсортированный по возрастанию массив, состоящий из целых случайных (псевдослучайных) чисел. Рядом с каждым числом выводится сумма всех предыдущих чисел. Далее программа выводит среднее значение массива и количество операций сравнения, которое потребовалось для сортировки массива. Прикреплены рисунки работы двух программ, с сортировкой пузырьком и сортировкой выбором с исходными данными: размер массива – 10, амплитуда чисел – 1000. (Рис. 1, Рис. 2)



Рис. 1 – Сортировка пузырьком

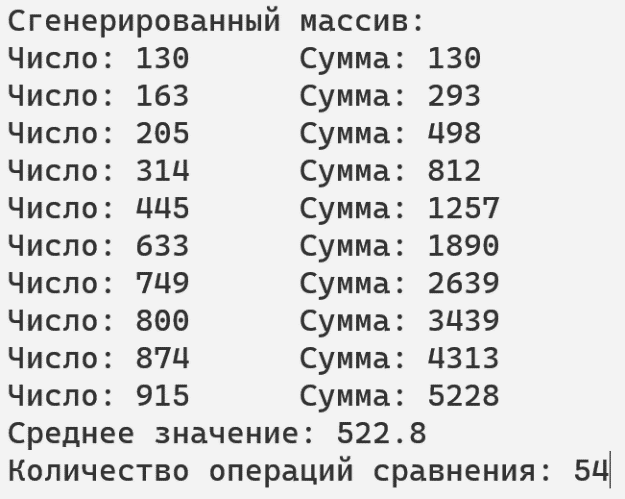


Рис. 2 – Сортировка выбором

**Пути дальнейшего улучшения программы**

1. Улучшение визуальной части программы
2. Создание более совершенного и эффективного метода сортировки массива

**Вывод**

По количеству операций сравнения (Сортировка – 39, выбор – 54) могу сделать вывод, что сортировка пузырьком эффективнее сортировки выбором.