

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Факультет систем управління літальних апаратів
Кафедра систем управління літальних апаратів

Лабораторна робота № 8

з дисципліни «Алгоритмізація та програмування»
на тему «Реалізація алгоритмів сортування та робота з файлами на мові C ++»

XAI.301.175.318.16ЛР

Виконав студент гр. 318

Нікуліна К.О.
(підпис, дата) (П.І.Б.)

Перевірив

к.т.н., доц. Олена ГАВРИЛЕНКО
(підпис, дата) (П.І.Б.)

МЕТА РОБОТИ

Вивчити теоретичний матеріал по алгоритмам обробки масивів на мові C++, а також бібліотеки для роботи з файлами і реалізувати оголошення, введення з файлу, обробку і виведення в файл одновимірних і двовимірних масивів на мові C++ в середовищі Visual Studio.

ПОСТАНОВКА ЗАДАЧІ

Завдання 1. За допомогою текстового редактору створити текстовий файл «array_in_71.txt» з елементами вихідного масиву. У програмі на C++ зчитати і перетворити цей масив відповідно до свого варіанту завдання (див. рис.1), ім'я файлу і необхідні змінні ввести з консолі. Вивести результати у файл «array_out_71.txt».

Array71 . | Дан масив розміру N. Поміняти порядок його елементів на зворотний.

Рис.1

Завдання 2. За допомогою текстового редактору створити текстовий файл «matr_in_36.txt» з елементами вихідного двовимірного масиву. У програмі зчитати і обробити матрицю відповідно до свого варіанту завдання (див. рис.2), ім'я файлу і необхідні змінні ввести з консолі. Дописати результати в той же файл.

Matrix36. Дана цілочисельна матриця розміру $M \times N$, елементи якої можуть набувати значень від 0 до 100. Різні рядки матриці назвемо схожими, якщо збігаються безлічі чисел, що зустрічаються в цих рядках. Знайти кількість рядків, схожих на перший рядок цієї матриці.

Рис.2

Завдання 3. Вивчити метод сортування відповідно до свого варіанту (див. рис.3), проаналізувати його складність і продемонструвати на прикладі з 7-ми елементів. Реалізувати у вигляді окремої функції алгоритм сортування елементів масиву. Зчитування і виведення відсортованого масиву організувати на файлах.

11 | Вставки

| Зростання

| Дійсний

Рис.3

Завдання 4. Введення, виведення, обробку масивів реалізувати окремими функціями з параметрами. Структурувати проєкт програми для виконання завдань 1-3 наступним чином:

```
main.cpp //основна функція і три функції для 3х завдань
array_utils.h //заголовки функцій для роботи з
               //одновимірними масивами (завдання1,3)
array_utils.cpp //визначення функцій для роботи з
               //одновимірними масивами (завдання1,3)
matrix_utils.h //заголовки функцій для роботи з
               //двовимірними масивами (завдання2)
matrix_utils.cpp //визначення функцій для роботи з
               //двовимірними масивами (завдання2)
```

ВИКОНАННЯ РОБОТИ

Завдання 1.

Вирішення задачі Array71

Вхідні дані (ім'я, опис, тип, обмеження):

числа, цілий тип.

Вихідні дані (ім'я, опис, тип):

cout «Порядковий номер» – порядковий номер унікального числа.

Алгоритм вирішення:

1. Ініціалізація;
2. Обмін елементів;
3. Завершення.

Завдання 2.

Вирішення задачі Matrix36.

Вхідні дані (ім'я, опис, тип, обмеження):

M - кількість рядків матриці, дійсний тип, >0 ;

N - кількість стовпців матриці, дійсний тип, >0 ;

matrix[i][j] – елемент матриці в рядку i, стовпці j, дійсний тип, $0 \leq \text{matrix}[i][j] \leq 100$.

Вихідні дані (ім'я, опис, тип):

count - кількість рядків схожих на перший, дійсний тип, $0 \leq \text{count} \leq M-1$

Алгоритм вирішення:

1. Перевірити, чи M та N більше 0. Якщо ні — вивести помилку.
2. Зчитати матрицю з консолі, перевіряючи, чи кожен елемент $\in [0, 100]$.
3. Якщо матриця порожня, вивести повідомлення.
4. Створити множину firstRowSet з елементів першого рядка.
5. Для кожного рядка i від 1 до M-1:
 - Створити множину currentRowSet з елементів поточного рядка.
 - Порівняти currentRowSet з firstRowSet.
 - Якщо множини ідентичні, збільшити count на 1.
6. Вивести count.

Завдання 3.

Вирішення задачі Sort11

Вхідні дані (ім'я, опис, тип, обмеження):

Масив з дійсних чисел, дійсний тип.

Вихідні дані (ім'я, опис, тип):

Відсортований масив (за зростанням), дійсний тип.

Алгоритм вирішення:

1. Починаємо з другого елемента;
2. Вибираємо поточний елемент;
3. Порівнюємо з елементами вже відсортованої частини (зліва від другого елемента) і зсуваємо їх праворуч, якщо вони більші за поточний.
4. Вставляємо на знайдену позицію.
5. Повторюємо для всіх елементів.

Лістинг коду вирішення задачі розділ і номер задач наведено в дод. А (стор. 7). Екран роботи програми показаний на рис. Б.1.

ВИСНОВКИ

Вивчено теоретичний матеріал з обробки масивів, бібліотеки для роботи з файлами, методи сортування масивів.

ДОДАТОК А

Лістинг коду програми

```
#include <iostream>
#include "array_utils.h"
#include "matrix_utils.h"
#include "sort_utils.h"

using namespace std;
int main()
{
    setlocale(LC_ALL, "rus");
    int choice;

    do
    {
        cout << "\nВыберите задание:\n"
              << "1. Array71\n"
              << "2. Matrix36\n"
              << "3. Sort11\n"
              << "0. Выход\n"
              << "Ваш выбор: ";
        cin >> choice;

        switch (choice)
        {
            case 1:
                array_71();
                break;
            case 2:
                matrix_36();
                break;
            case 3:
                sort_11();
                break;
            case 0:
                cout << "Выход из программы.\n";
                break;
            default:
                cout << "Неверный выбор! Попробуйте еще раз.\n";
        }
    } while (choice != 0);

    return 0;
}
```


ДОДАТОК Б

Скріншоти вікна виконання програми

```
Выберите задание:
1. Array71
2. Matrix36
3. Sort11
0. Выход
Ваш выбор: 1

=== Задание 1: Array ===
Массив в обратном порядке записан в файл array_out_71.txt

Выберите задание:
1. Array71
2. Matrix36
3. Sort11
0. Выход
Ваш выбор: 2

=== Задание 2: Matrix36 ===
Количество строк, похожих на первую: 2
Результат дописан в тот же файл matr_in_36.txt

Выберите задание:
1. Array71
2. Matrix36
3. Sort11
0. Выход
Ваш выбор: 3

=== Задание 3: Sort11 ===
Сортировка завершена. Результат в файле sort_out_11.txt

Выберите задание:
1. Array71
2. Matrix36
3. Sort11
0. Выход
Ваш выбор: 0
Выход из программы.
```

Рисунок Б.1 – Экран виконання програми для вирішення завдання Array71, Matrix36, Sort11

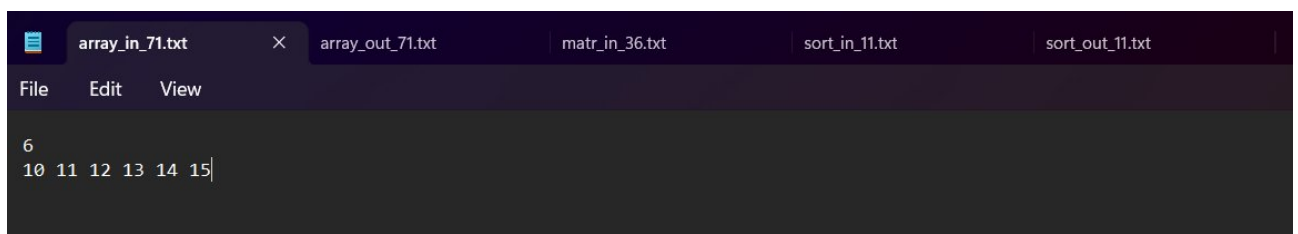


Рисунок Б.2 – Вхідні данні для вирішення завдання Array71

```

6
15 14 13 12 11 10

```

Рисунок Б.3 – Вихідні данні після вирішення завдання Array71

```

5 6
1 2 3 4 5 6
6 5 4 3 2 1
10 11 12 13 14 15
1 2 3 4 5 6
99 98 97 96 95 94
Количество строк, похожих на первую: 2

```

Рисунок Б.4 – Вхідні та вихідні данні для вирішення завдання Matrix36

```

10 5 11 6 7 1 4

```

Рисунок Б.5 – Вхідні данні для вирішення завдання Sort11

```

1 4 5 6 7 10 11

```

Рисунок Б.6 – Вихідні данні після вирішення завдання Sort11