

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет

Факультет систем управління літальних апаратів

Кафедра систем управління літальних апаратів

XAI.319. G7 Автоматизація, комп'ютерно-інтегровані технології та
робототехніка. 319. 38 ЛР

Виконав студент група 319

Олександр *Ткачук*

13.12.2025

Лабораторна робота №6 (методичні вказівки)

з дисципліни «Алгоритмізація та програмування»

Тема: "Реалізація алгоритмів обробки одновимірних масивів мовою C ++"

МЕТА РОБОТИ

Вивчити теоретичний матеріал з основ представлення одновимірних і масивів на мові C ++ і реалізувати декларацію, введення з консолі, обробку і виведення в консоль одновимірних масивів на мові C ++ в середовищі QtCreator.

ПОСТАНОВКА ЗАДАЧІ

Завдання 1. Вирішити завдання на аналіз і виведення елементів одновимірного масиву. Розмір масиву і його елементи ввести з консолі. Спершу вивести весь масив у рядок в порядку зростання індексів, потім – елементи чи підраховані результати відповідно до завдання. Варіанти завдань представлено в табл.1.

Завдання 2. Вирішити завдання на перетворення одновимірного масиву.

Розмір масиву і його елементи ввести з консолі. Спершу вивести у консоль заданий масив, потім – змінений. Варіанти завдань представлено в табл.2.

Завдання 3. У функції main() організувати багаторазовий вибір одного з двох завдань. Кожне завдання описати окремою функцією без параметрів. Введення, виведення, обробку масивів реалізувати окремими функціями з параметрами.

Завдання 4. Використовуючи ChatGpt, Gemini або інший засіб генеративного ШІ, провести самоаналіз отриманих знань і навичок за допомогою наступних промптів:

1) «Ти - викладач, що приймає захист моєї роботи. Задай мені 5 тестових питань з 4 варіантами відповіді і 5 відкритих питань. Це мають бути завдання <середнього> рівня складності на розвиток критичного та

інженерного мислення. Питання мають відноситись до коду, що є у файлі звіту, і до теоретичних відомостей, що є у файлі лекції»

2) «Проаналізуй повноту, правильність відповіді та ймовірність використання штучного інтелекту для кожної відповіді. Оціни кожне питання

2

у 5-балльній шкалі, віднімаючи 60% балів там, де ймовірність відповіді з засобом ШІ висока. Обчисли загальну середню оцінку»

3) «Проаналізуй код у звіті, і додай опис і приклади коду з питань, які є в теоретичних відомостях, але не відпрацьовано у коді при вирішенні завдань»

Проаналізуйте задані питання, коментарі і оцінки, надані ШІ. Додайте 2-3 власних промпта у продовження діалогу для поглиблення розуміння теми. Рекомендації до виконання завдань 1-2.

Умова виду «дан масив розміру N» означає, що спочатку вводиться фактичний розмір масиву (ціле число N), а потім - всі його елементи. Якщо в завданні явно не вказується, які значення може приймати розмір вихідного масиву, то передбачається, що розмір може змінюватися в межах від 2 до 50.

Порядковий номер початкового елемента масиву в постановці завдання вважається рівним 1. Якщо не вказано іншого, тип елементів – дійсний.

Якщо в завданні не описаний результуючий набір даних, то передбачається, що цим набором є створений (перетворений) масив, і необхідно вивести в строку всі його елементи в порядку зростання їх індексів.

При виконанні завдань на перетворення масиву не слід використовувати допоміжні масиви.

У звіті повинні бути вхідні і вихідні дані для кожного завдання, лістинг коду програми та скріншот екрану виконання. Код повинен містити КОМЕНТАРИ!!! Крім того, в звіті повинні бути представлені три діаграми

активності: для функції аналізу елементів завдання 1, для функції модифікації масиву завдання 2 і для завдання 3.

Array48.	Дан ціличесельний масив розміру N. Знайти максимальну кількість його одинакових елементів.
Array101.	Дан масив розміру N і ціле число K ($1 \leq K \leq N$). Перед елементом масиву з порядковим номером K вставити новий елемент з нульовим значенням.

Лістинг коду програми

```
// Лабораторна робота

// Тема: Реалізація алгоритмів обробки одновимірних масивів мовою C++

// Середовище: QtCreator (консольна програма)

#include <iostream>
#include <limits>
#include <string>
#include <unordered_map>

using namespace std;

// Максимальний розмір масиву
const int N = 50;

// ===== ВВЕДЕННЯ ЧИСЕЛ =====

// Функція для введення додатного цілого числа (>0)
int input_positive_int(const string& prompt) {
    int value;
    while (true) {
        cout << prompt;
        cin >> value;

        // Перевірка на помилкове введення або недодатне число
        if (cin.fail() || value <= 0) {
            cin.clear(); // очищення пропорців помилки
            cin.ignore(numeric_limits<streamsize>::max(), '\n'); // очищення буфера
            cout << "Помилка! Введіть ціле число більше 0.\n";
        }
    }
}
```

```
    } else {

        return value;
    }
}

}

// Функція для введення будь-якого цілого числа

int input_int(const string& prompt) {

    int value;

    while (true) {

        cout << prompt;
        cin >> value;

        // Перевірка на некоректне введення

        if (cin.fail()) {

            cin.clear();

            cin.ignore(numeric_limits<streamsize>::max(), '\n');

            cout << "Помилка! Введіть ціле число.\n";

        } else {

            return value;
        }
    }
}

// Функція введення цілого числа в заданому діапазоні

int input_int_range(const string& prompt, int min, int max) {

    int value;

    while (true) {

        value = input_int(prompt);

        if (value < min || value > max) {

            cout << "Помилка! Введіть число в діапазоні ["
                << min << ";" << max << "].\n";

        } else {

            return value;
        }
    }
}
```

```

}

// ===== РОБОТА З МАСИВОМ =====

// Функція введення масиву

void input_array(int arr[N], int& n) {
    // Введення кількості елементів масиву

    while (true) {

        n = input_positive_int("Count of elements (1..50): ");

        if (n <= N) break;

        cout << "Помилка! Максимальний розмір масиву - " << N << ".\n";
    }
}

// Введення елементів масиву

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

    arr[i] = input_int("Element " + to_string(i + 1) + ": ");
}

}

// Функція виведення масиву

void output_array(const int arr[N], int n) {
    cout << "Array elements:\n";

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

        cout << arr[i] << " ";
    }

    cout << "\n";
}

}

// ===== ЗАВДАННЯ Array48 =====

// Знайти максимальну кількість однакових елементів у масиві


int max_equal_count(const int arr[N], int n) {
    unordered_map<int, int> freq; // контейнер для підрахунку частот

    int maxCount = 0;

    // Підрахунок кількості кожного елемента
}
```

```

for (int i = 0; i < n; i++) {
    freq[arr[i]]++;
    if (freq[arr[i]] > maxCount) {
        maxCount = freq[arr[i]];
    }
}

return maxCount;
}

// Функція виконання завдання Array48
void task_array48() {
    int arr[N];
    int n;

    input_array(arr, n);      // введення масиву
    output_array(arr, n);     // виведення масиву

    int result = max_equal_count(arr, n);
    cout << "Maximum count of equal elements: " << result << "\n";
}

// ===== ЗАВДАННЯ Array101 =====
// Перед елементом з номером K вставити елемент зі значенням 0

bool insert_zero_before_k(int arr[N], int& n, int k) {
    // Перевірка на переповнення масиву
    if (n >= N) return false;

    // Зсув елементів вправо
    for (int i = n; i >= k; i--) {
        arr[i] = arr[i - 1];
    }

    // Вставка 0 перед K-м елементом
    arr[k - 1] = 0;
}

```

```

n++; // збільшення розміру масиву

return true;
}

// Функція виконання завдання Array101
void task_array101() {
int arr[N];
int n;

input_array(arr, n); // введення масиву

cout << "\nOriginal array:\n";
output_array(arr, n); // виведення початкового масиву

// Введення номера елемента K
int k = input_int_range("K (1..n): ", 1, n);

// Вставка елемента
if (!insert_zero_before_k(arr, n, k)) {
cout << "Помилка! Масив заповнений.\n";
return;
}

cout << "\nModified array (0 inserted before element #" << k << "):\n";
output_array(arr, n);
}

// ====== МЕНЮ ======
// Виведення меню
void show_menu() {
cout << "\n===== MENU =====\n";
cout << "1 - Array48 (max frequency of equal elements)\n";
cout << "2 - Array101 (insert 0 before element K)\n";
cout << "0 - Exit\n";
}

```

```
cout << "=====\\n";
}

// ===== ГОЛОВНА ФУНКЦІЯ =====

int main() {
    ios::sync_with_stdio(false);
    cin.tie(nullptr);

    while (true) {
        show_menu();
        int choice = input_int("Choose task: ");

        if (choice == 0) {
            cout << "Program finished.\\n";
            break;
        }

        switch (choice) {
            case 1:
                task_array48();
                break;
            case 2:
                task_array101();
                break;
            default:
                cout << "Невірний пункт меню.\\n";
        }
    }
}

return 0;
```

}

----- MENU -----

- 1 - Array48 (max frequency of equal elements)
 - 2 - Array101 (insert 0 before element K)
 - 0 - Exit
-

Choose task: 2

Count of elements (1..50): 5

Element 1: 2

Element 2: 5

Element 3: 2

Element 4: 5

Element 5: 3

Original array:

Array elements:

2 5 2 5 3

K (1..n): 6

Error! Enter a value in range [1; 5].

K (1..n): 3

Modified array (0 inserted before element #3):

Array elements:

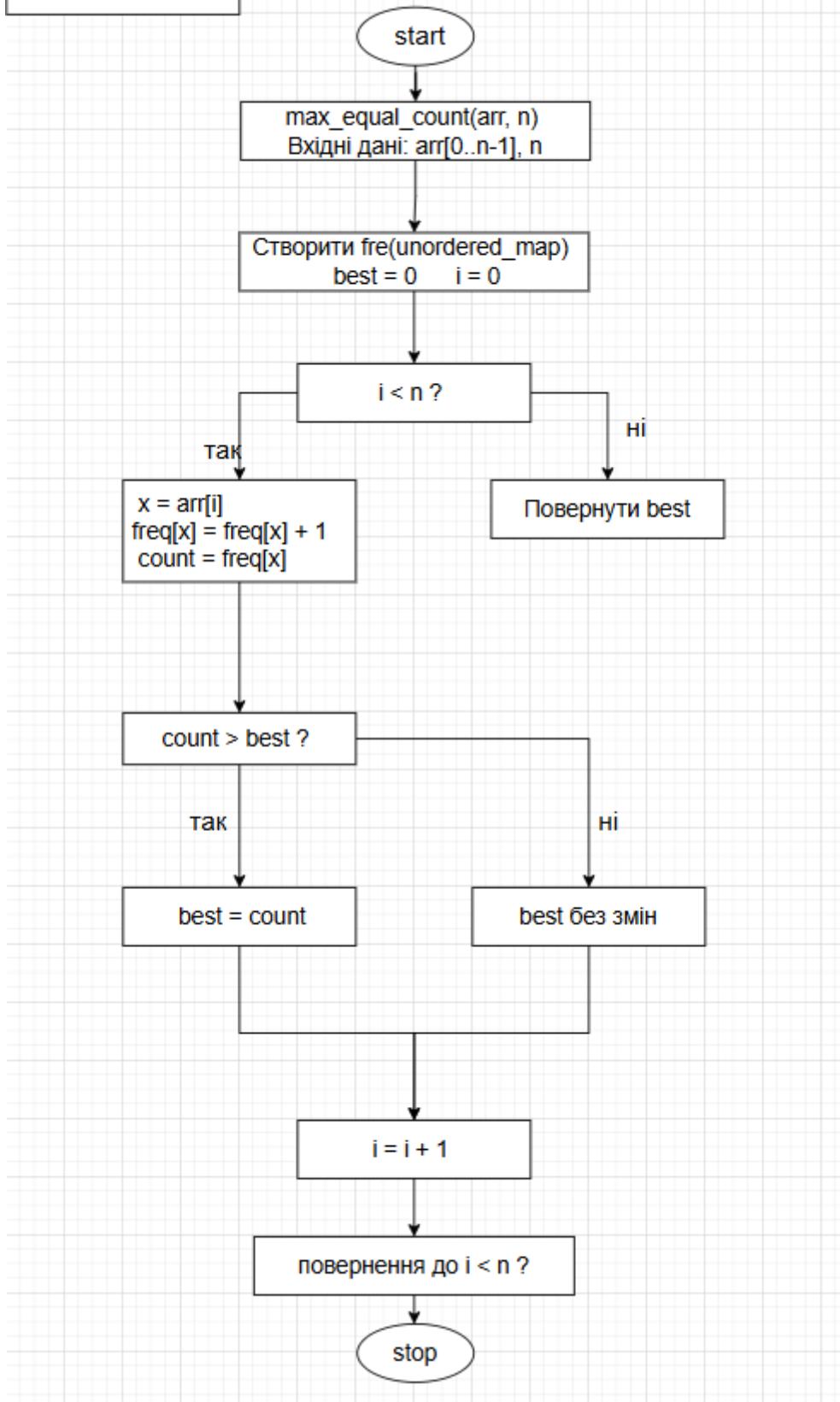
2 5 0 2 5 3

----- MENU -----

- 1 - Array48 (max frequency of equal elements)
 - 2 - Array101 (insert 0 before element K)
 - 0 - Exit
-

Choose task:

Array48



array101

start

- введення масиву $a[0..n-1]$
-виведення початкового масиву
-введення номера K
-виклик `insert_zero_before_k(a,n,K)`
-виведення зміненого масиву

`insert_zero_before_k(arr, n, k)`
Вхідні дані: $arr[0..n-1]$, n , k

$n \geq N ?$

так

ні

Повернути false
(масив переповнений)

$i = n$

$i \geq k ?$

так

$arr[i] = arr[i-1]$
 $i = i - 1$

коли ні

$arr[k-1] = 0$
 $n = n + 1$
Повернути true

повернення до $i \geq k ?$

Stop



<https://chatgpt.com/share/693f11f5-64e0-8012-9d11-22002aad15c0>