

**Звіт**  
**Оскорбіна Марія 117 група**  
**Лабораторна робота 5**

1. Підключаємо бібліотеки

```
#include <iostream>
#include <cstdlib>
#include <ctime>
#include <locale.h>
#include <windows.h>
#include <conio.h>
```

2. Константа, яка визначає максимальний розмір масиву

```
const int MAX_SIZE = 100;
```

3. void тут використовується для заповнення масиву випадковими значеннями в заданому діапазоні

```
void generateArray(int arr[], int size, int rangeStart, int rangeEnd) {
    std::srand(static_cast<unsigned int>(std::time(nullptr)));
```

```
    for (int i = 0; i < size; ++i) {
        arr[i] = std::rand() % (rangeEnd - rangeStart + 1) + rangeStart;
    }
}
```

4. void призначений для пошуку індексів максимального негативного та мінімального позитивного елементів у масиві.

```
void findMinMaxIndices(int arr[], int size, int& minPositiveIndex, int&
maxNegativeIndex) {
```

```
    minPositiveIndex = -1;
    maxNegativeIndex = -1;

    for (int i = 0; i < size; ++i) {
        if (arr[i] > 0 && (minPositiveIndex == -1 || arr[i] <
arr[minPositiveIndex])) {
            minPositiveIndex = i;
        }
        else if (arr[i] < 0 && (maxNegativeIndex == -1 || arr[i] >
arr[maxNegativeIndex])) {
            maxNegativeIndex = i;
        }
    }
}
```

5. Основна функція, яка виконує сортування масивів

**int main()**

6. Створюємо змінну, яка буде використовуватися для збереження вибору користувача, тобто виведе меню і користувач обере що йому треба

**int choice;**

**std::cout << "Choose the initialization method:\n"**

**<< "1. Manual input\n"**

**<< "2. Random generation\n";**

**std::cin >> choice;**

**7.switch (choice) {**

**case 1:**

**inputArray(arr, size);**

**break;**

**case 2: {**

**int rangeStart, rangeEnd;**

**std::cout << "Enter the range for random generation (start end): ";**

**std::cin >> rangeStart >> rangeEnd;**

**generateArray(arr, size, rangeStart, rangeEnd);**

**break;**

**}**

**default:**

**std::cerr << "Invalid choice.\n";**

**return 1;**

**}**

switch (choice): Це ключове слово switch, яке починає конструкцію перевірки на рівні значень.

case 1: Це мітка для першого випадку. Якщо значення choice дорівнює 1, програма виконає код, який знаходиться під цією міткою.

inputArray(arr, size);: Викликає функцію inputArray, яка, очевидно, використовується для введення масиву вручну.

break;: Завершує виконання коду для цього випадку і переходить за межі switch.

case 2:: Це мітка для другого випадку. Якщо значення choice дорівнює 2, програма виконає код, який знаходиться під цією міткою.

int rangeStart, rangeEnd;: Оголошує змінні rangeStart і rangeEnd.

`std::cout << "Enter the range for random generation (start end): ";` Виводить повідомлення, яке запрошує користувача ввести діапазон для випадкової генерації.

`std::cin >> rangeStart >> rangeEnd;` Зчитує значення `rangeStart` і `rangeEnd` зі стандартного вводу.

`break;` Завершує виконання коду для цього випадку і переходить за межі `switch`.

`default::` Це мітка, яка використовується для визначення коду, який буде виконуватися в тому випадку, якщо `choice` не відповідає жодній з міток `case`.

`std::cerr << "Invalid choice.\n";` Виводить повідомлення про недійсний вибір у потік помилок.

`return 1;` Припиняє виконання програми і повертає значення 1, що може вказувати на помилку.