Міністерство освіти і науки України

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського"

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 8 з дисципліни «Алгоритми та структури даних-1. Основи алгоритмізації»

«Дослідження алгоритмів пошуку та сортування»

Варіант 16

Виконав студент ІП-11 Кузьменков Дмитро Олегович

(шифр, прізвище, ім'я, по батькові)

Перевірив Мартинова. О. П

( прізвище, ім'я, по батькові)

Київ 2021

**Лабораторна робота 8**

**Дослідження алгоритмів пошуку та сортування**

Мета – дослідити алгоритми пошуку та сортування, набути практичних навичок використання цих алгоритмів під час складання програмних специфікацій.

# Завдання:

1. Опису змінної індексованого типу (двовимірний масив) згідно з варіантом.

2. Ініціювання змінної, що описана в п.1 даного завдання.

3. Створення нової змінної індексованого типу (одновимірний масив) та її ініціювання значеннями, що обчислюються згідно з варіантом.



**Постановка задачі:**

Опишемо двомірний масив, задаємо його розмірність відповідно до завдання. Ініціюємо його випадковими числами за допомогою srand() та вкладеного циклу. Ділі знаходимо середнє арифметичне усіх від’ємних елементів кожного стовпця за допомогою вкладеного циклу та за допомогою отриманих значень, ініціюємо два одновимірні масиви. Другий масив відсортовуємо за спаданням методом обміну (SortArray()). Виводимо двомірний масив, перший одновимірний масив (невідсортований) та другий одновимірний масив (відсортований).

# Побудова математичної моделі. Складемо таблицю імен змінних:

|  |  |  |  |
| --- | --- | --- | --- |
| Змінна | Тип | Ім’я | Призначення |
| Двовимірний масив | Двовимірний масив з цілими елементами | Matrix[][] | Початкове дане |
| Перший одновимірний масив (невідсортований) | Масив з дійсними елементами | Array[] | Проміжне дане |
| Другий одновимірний масив (невідсортований) | Масив з дійсними елементами | Array2[] | Проміжне дане, результат |
| Лічильник циклу, індекс елемента масиву | Натуральне | i | Проміжне дане |
| Лічильник циклу, індекс елемента масиву | Натуральне | j | Проміжне дане |
| Кількість від’ємних елементів у стовпчику | Натуральне | y | Проміжне дане |
| Сума від’ємних елементів стовпчика | Ціле | ColumnSum | Проміжне дане |
| Середнє арифметичне | Дійсне | ArithmeticMean | Проміжне дане |

Крок 1. Деталізуємо складний цикл, що ініціює Matrix[5][7]. Деталізуємо складний цикл, що знаходить середнє арифметичне усіх від’ємних елементів кожного стовпця та ініціює Array[7] та Array2[7].

Крок 2. Деталізуємо функцію сортування масиву методом обміну за спаданням.

Виводимо Matrix[5][7], Array[7], Array2[7].

1. **Псевдокод:**

# Початок

1) Деталізуємо складний цикл, що ініціює Matrix[5][7]. Деталізуємо складний цикл, що знаходить середнє арифметичне усіх від’ємних елементів кожного стовпця та ініціює Array[7] та Array2[7]

2) Деталізуємо функцію сортування масиву методом обміну за спаданням.

Виводимо Matrix[5][7], Array[7], Array2[7]

**Кінець**

*Крок1*

# Початок

1) **Поки** i:= 0; i < 5; i++;

**Повторити**

**Поки** j:= 0; j < 7; i++;

**Повторити**

Matrix[i][j] = rand() % 100 – 50;

**Все повторити**

**Все повторити**

**Поки** j:= 0; i < 7; j++;

**Повторити**

y:= 0;

ColumnSum:= 0;

**Поки** i:= 0; i < 5; i++;

**Повторити**

**Якщо** Matrix[i][j] < 0

**То** ColumnSum:+= Matrix[i][j];

y++;

**Все повторити**

ArithmeticMean:= ColumnSum / y;

Array[j]:= ArithmeticMean;

Array2[j]:= ArithmeticMean;

**Все повторити**

SortArray(Array2[], 7)

Виводимо Matrix[5][7], Array[7], Array2[7]

2) Деталізуємо функцію сортування масиву методом обміну за спаданням;

**Кінець**

*Крок2*

**SortArray(Array[], size):=**

i:= 0;

**Поки** i < size - 1 **або** Array[i] != 0

**Повторити**

j:= 0;

**Поки** j < size – 1 - i **або** Array[j] != 0

**Повторити**

**Якщо** Array[j + 1] > Array[j]

**То**

tmp = Array[j];

Array[j] = Array[j + 1];

Array[j + 1] = tmp;

j++;

**Все повторити**

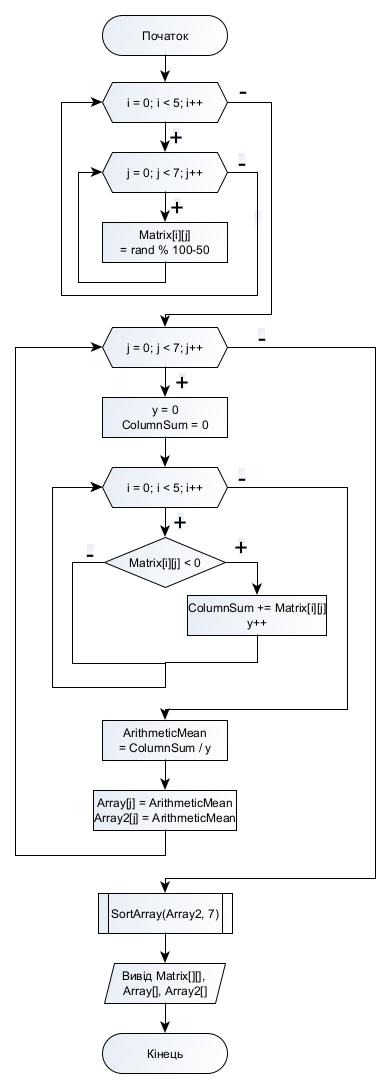
i++;

**Все повторити**

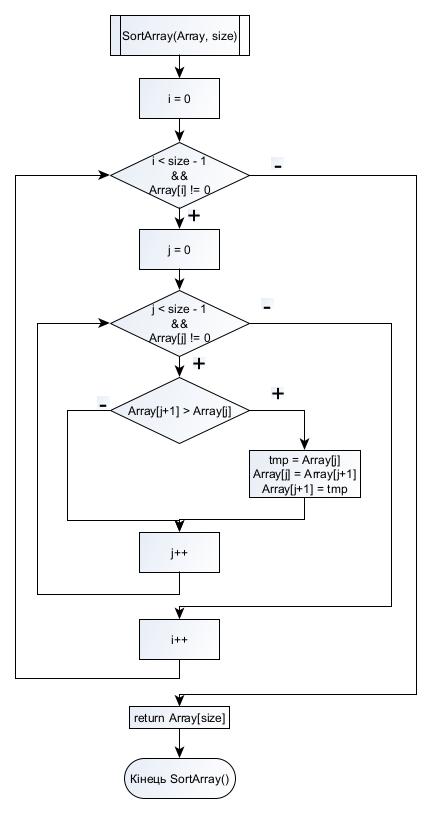
**Return** Array[size]

1. **Блок-схема**

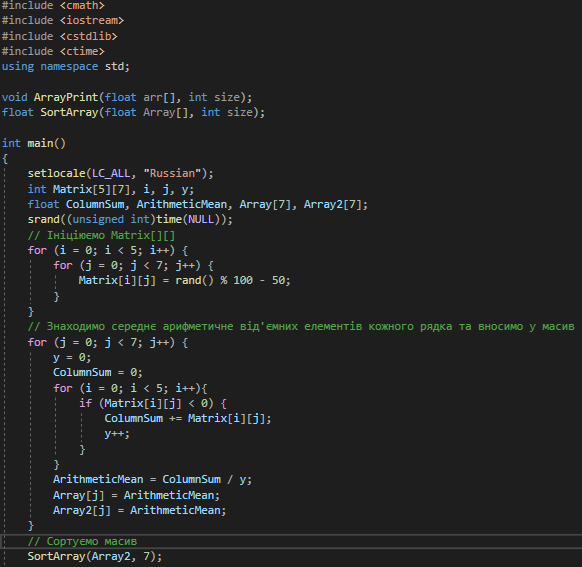
*Крок1*

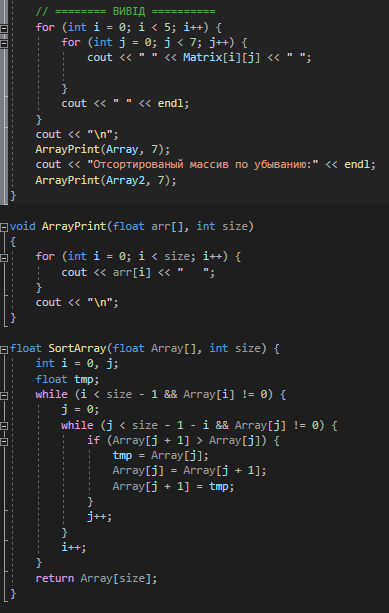


*Крок2*

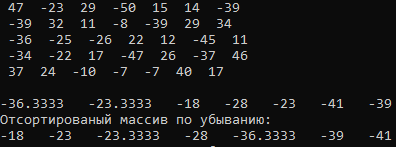


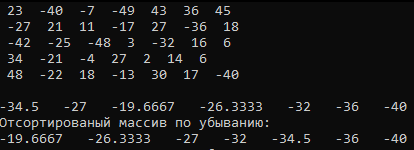
**4. Код (C++):**

****

****

**Результати:**

****

****

**Висновок:** Виконуючи лабораторну роботу я дослідив алгоритми пошуку та сортування, набув практичних навичок використання цих алгоритмів під час складання програмних специфікацій. Побудуваа алгоритм та написав програму, що виводить середнє арифметичне від’ємних елементів кожного стовпця заданої матриці та сортує отримані значення обміном за спаданням.