

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 5

Виконав студент

ІП-13 Вальчишен Ярослав Олександрович
(шифр, прізвище, ім'я, по батькові)

Перевірів

Вечерковська А.С.
(прізвище, ім'я, по батькові)

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.

Варіант 6.

Розробити алгоритм та написати програму, яка складається з наступних дій:

1. Опис змінної індексованого типу (двовимірний масив).
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 (табл. 1).

5	5 x 7	Цілий	Із суми значень елементів стовпців двовимірного масиву. Відсортувати методом бульбашки за зростанням.
---	-------	-------	---

1. Постановка задачі

Результатом розв'язку є масив, який складається з суми значень елементів стовпців двовимірного масиву, відсортований методом бульбашки за спаданням.

2. Побудова математичної моделі

Таблиця змінних

Змінна	Тип	Ім'я	Призначення
Двовимірний масив	Цілий	matrix	Проміжне дане
Лічильник циклу	Натуральний	i	Проміжне дане
Лічильник вкладеного циклу	Натуральний	j	Проміжне дане
Масив	Цілий	array	Вихідні дані

Таблиця функцій

Назва	Синтаксис	Призначення
Генерація випадкового значення	Next(a,b)	Генерує випадкове число від a до b невиключно
Отримання довжини масиву	Length(array)	Повертає довжину масиву array

3. Розв'язання

Програмні специфікації запишемо у псевдокодї та графічній формї у вигляді блок-схеми.

- Крок 1. Визначимо основні дії
Крок 2. Деталізуємо дію оголошення змінних
Крок 3. Деталізуємо дію заповнення матриці випадковими значеннями
Крок 4. Деталізуємо дію заповнення масиву з суми значень елементів стовпців двовимірного масиву
Крок 5. Деталізуємо дію сортування масиву
Крок 6. Деталізуємо дію виведення масиву

Псевдокод

Крок 1

Підпрограма FillMatrix(matrix)

Заповнення матриці випадковими значеннями

Все підпрограма

Підпрограма FillArray(array, matrix)

Заповнення масиву з суми значень елементів стовпців двовимірного масиву

Все підпрограма

Підпрограма Sort(array)

Сортування масиву

Все підпрограма

Підпрограма DisplayArray(array)

Виведення масиву

Все підпрограма

початок

Оголошення змінних

Виклик функції FillMatrix(matrix)

Виклик функції FillArray(array, matrix)

Виклик функції Sort(array)

Виклик функції DisplayArray(array)

кінець

Крок 2

Підпрограма FillMatrix(matrix)

Заповнення матриці випадковими значеннями

Все підпрограма

Підпрограма FillArray(array, matrix)

Заповнення масиву з суми значень елементів стовпців двовимірного масиву

Все підпрограма

Підпрограма Sort(array)

Сортування масиву

Все підпрограма

Підпрограма DisplayArray(array)

Виведення масиву

Все підпрограма

Крок 3

Підпрограма FillMatrix(matrix)

для i від 0 до 5 повторити

для j від 0 до 7 повторити

matrix[i, j] = Next(-100, 101)

все повторити

все потворити

Все підпрограма

Підпрограма FillArray(array, matrix)

Заповнення масиву з суми значень елементів стовпців двовимірного масиву

Все підпрограма

Підпрограма Sort(array)

Сортування масиву

Все підпрограма

Підпрограма DisplayArray(array)

Виведення масиву

Все підпрограма

початок

int[,] matrix = new int[5, 7]

int[] array = new int[7]

Виклик функції FillMatrix(matrix)

Виклик функції FillArray(array, matrix)

Виклик функції Sort(array)

Виклик функції DisplayArray(array)

кінець

Крок 4

Підпрограма FillMatrix(matrix)

для i від 0 до 5 повторити

для j від 0 до 7 повторити

matrix[i, j] = Next(-100, 101)

все повторити

все потворити

Все підпрограма

Підпрограма FillArray(array, matrix)

для і від 0 до 7 повторити

для j від 0 до 5 повторити

array[i] = array[i] + matrix[j, i]

все повторити

все потворити

Все підпрограма

Підпрограма Sort(array)

Сортування масиву

Все підпрограма

Підпрограма DisplayArray(array)

Виведення масиву

Все підпрограма

початок

int[,] matrix = new int[5, 7]

int[] array = new int[7]

Виклик функції FillMatrix(matrix)

Виклик функції FillArray(array, matrix)

Виклик функції Sort(array)

Виклик функції DisplayArray(array)

кінець

Крок 5

Підпрограма FillMatrix(matrix)

для і від 0 до 5 повторити

для j від 0 до 7 повторити

matrix[i, j] = Next(-100, 101)

все повторити

все потворити

Все підпрограма

Підпрограма FillArray(array, matrix)

для і від 0 до 7 повторити

для j від 0 до 5 повторити

array[i] = array[i] + matrix[j, i]

все повторити

все потворити

Все підпрограма

Підпрограма Sort(array)

tmp = 0

для i від 0 до Length(array) **повторити**

для j від 0 до Length(array) - 1 **повторити**

якщо array[j] > array[i + 1]

то

tmp = array[j]

array[j] = array[j + 1]

array[j + 1] = tmp

все якщо

все повторити

все потворити

Все підпрограма

Підпрограма DisplayArray(array)

Виведення масиву

Все підпрограма

початок

int[,] matrix = new int[5, 7]

int[] array = new int[7]

Виклик функції FillMatrix(matrix)

Виклик функції FillArray(array, matrix)

Виклик функції Sort(array)

Виклик функції DisplayArray(array)

кінець

Крок 6

Підпрограма FillMatrix(matrix)

для i від 0 до 5 **повторити**

для j від 0 до 7 **повторити**

matrix[i, j] = Next(-100, 101)

все повторити

все потворити

Все підпрограма

Підпрограма FillArray(array, matrix)

для i від 0 до 7 **повторити**

для j від 0 до 5 **повторити**

array[i] = array[i] + matrix[j, i]

все повторити

все потворити

Все підпрограма

Підпрограма Sort(array)

tmp = 0

для i від 0 до Length(array) **повторити**

для j від 0 до Length(array) - 1 **повторити**

якщо array[j] > array[i + 1]

то

tmp = array[j]

array[j] = array[j + 1]

array[j + 1] = tmp

все якщо

все повторити

все потворити

Все підпрограма

Підпрограма DisplayArray(array)

для i від 0 до Length(array) повторити

Виведення array[i]

Все підпрограма

початок

int[,] matrix = new int[5, 7]

int[] array = new int[7]

Виклик функції FillMatrix(matrix)

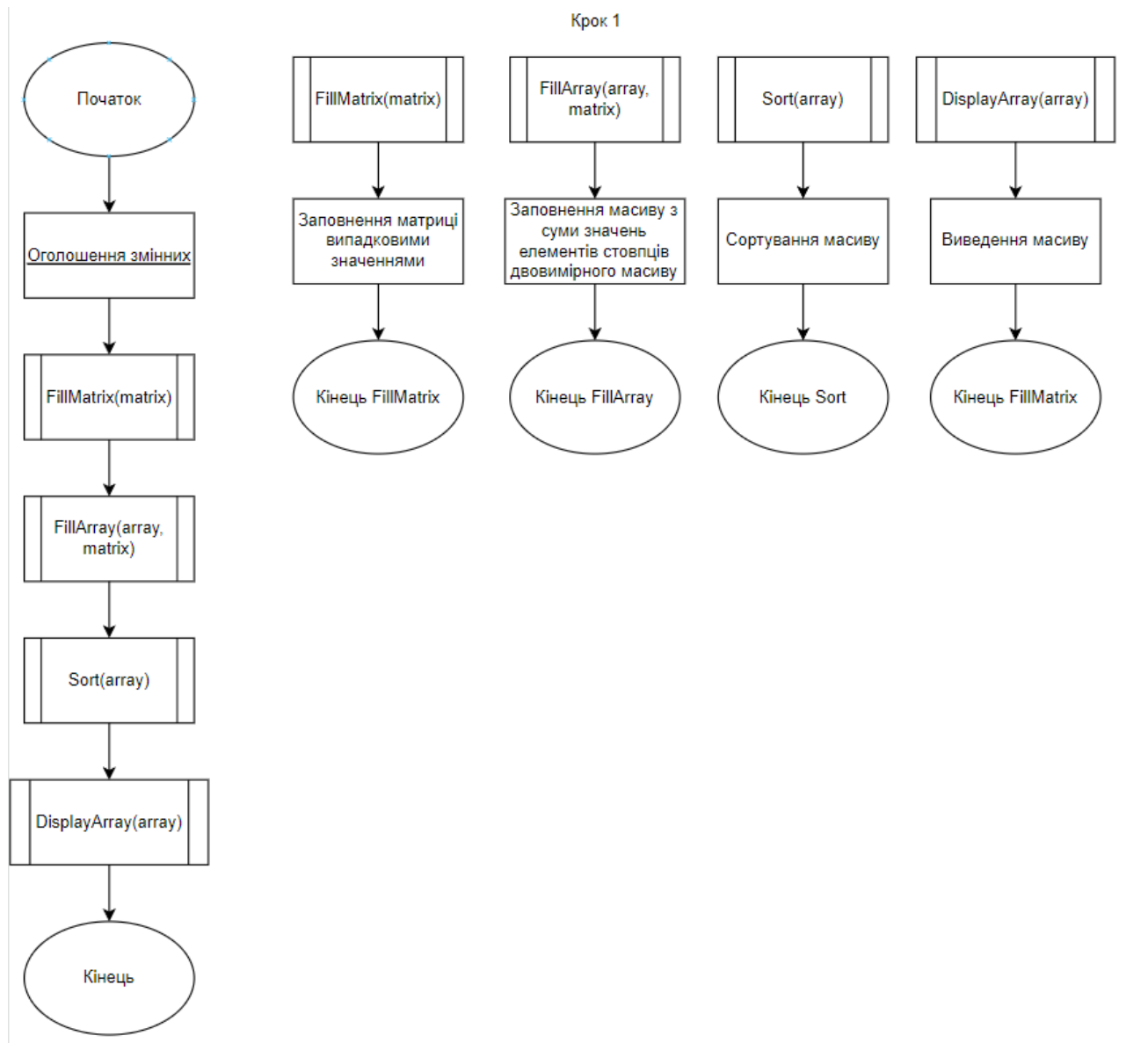
Виклик функції FillArray(array, matrix)

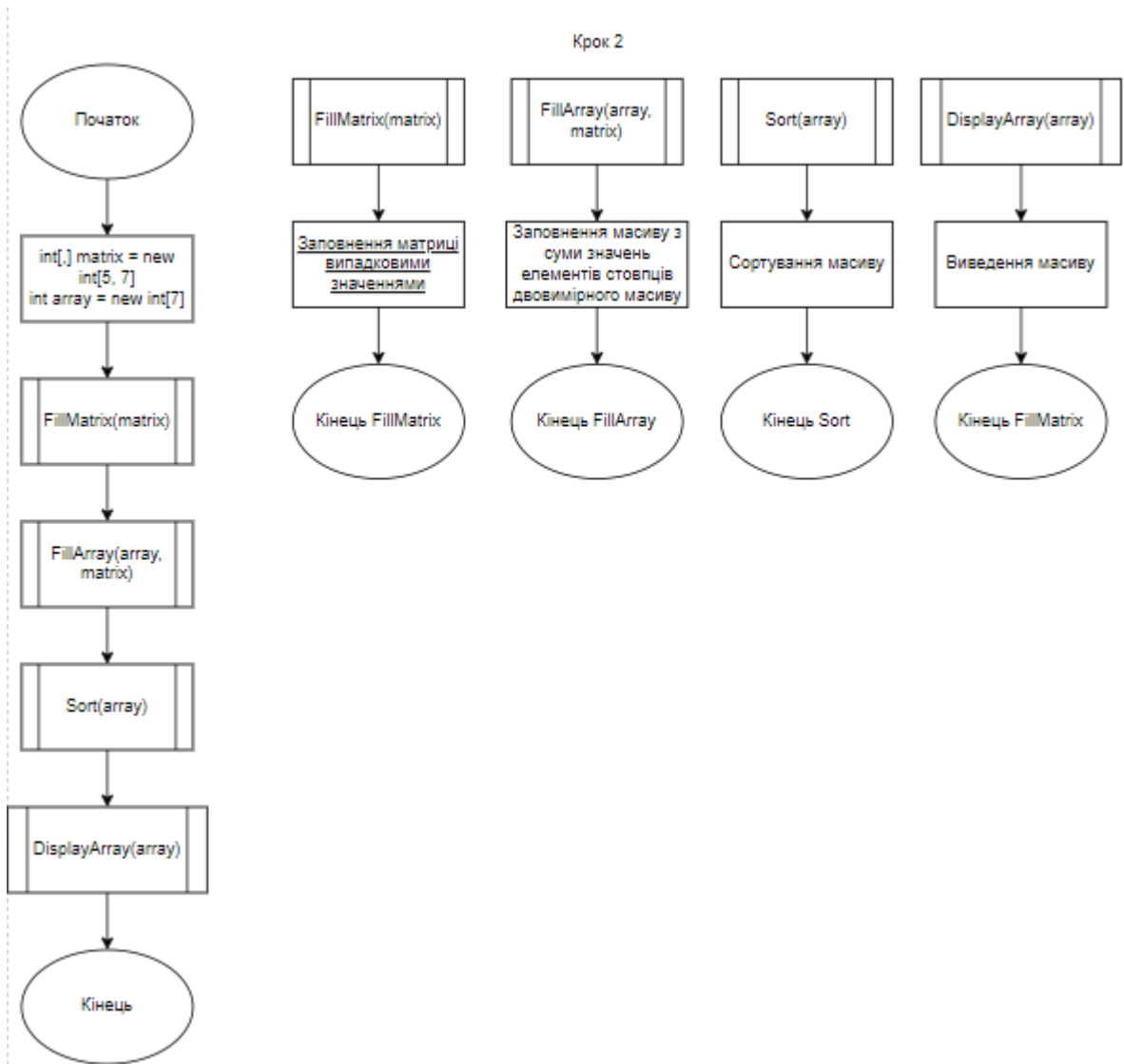
Виклик функції Sort(array)

Виклик функції DisplayArray(array)

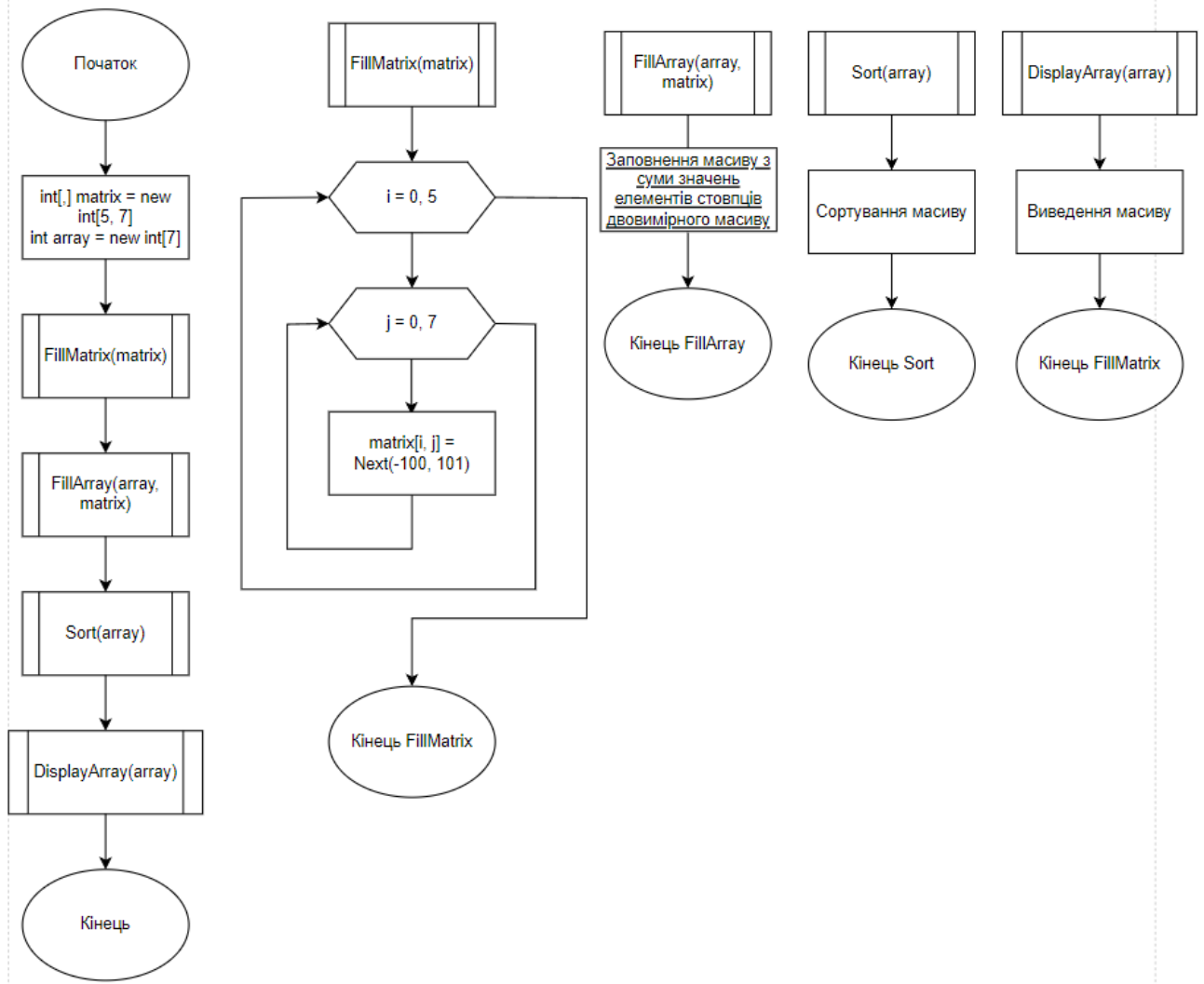
кінець

Блок-схема

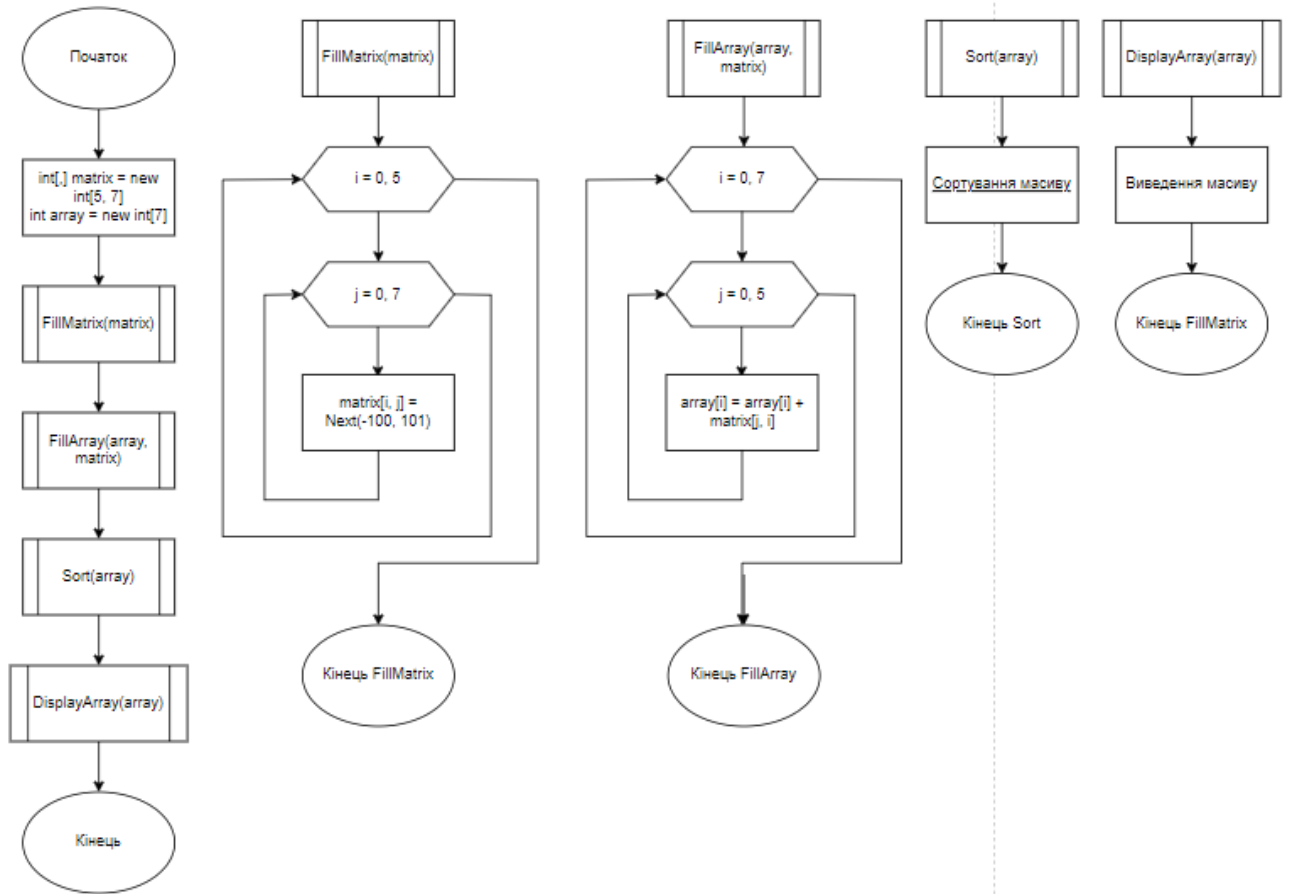




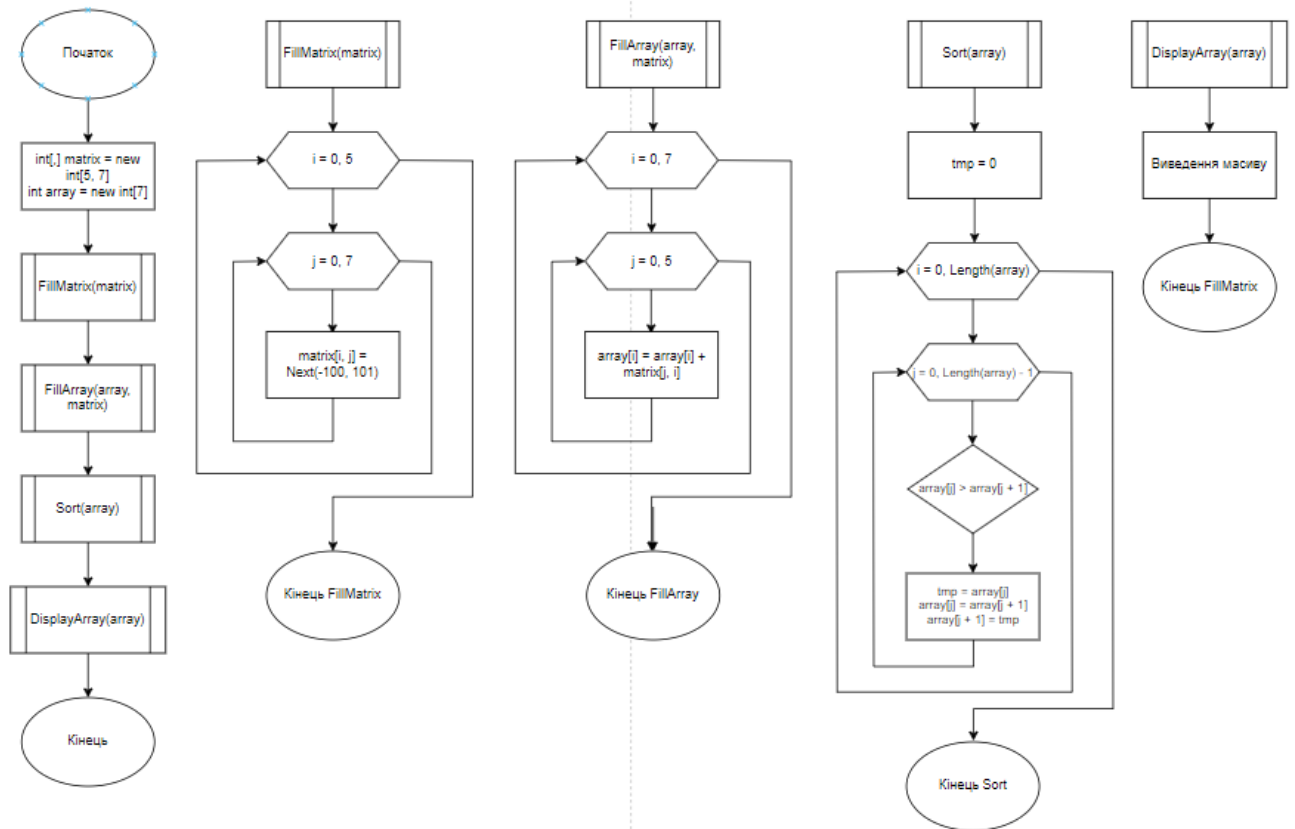
Крок 3

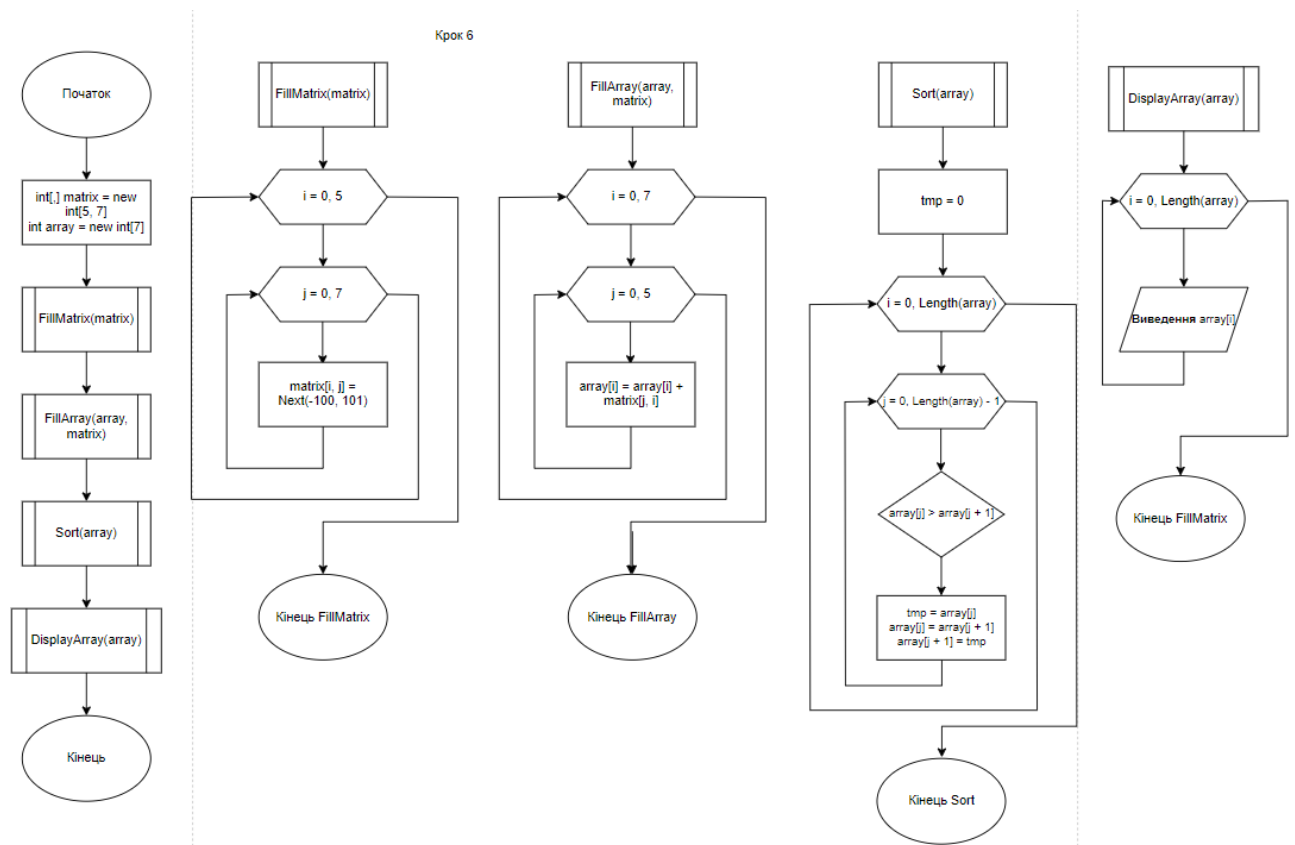


Крок 4



Крок 5





Код програми

```

class Program
{
    Ссылка: 0
    static void Main(string[] args)
    {
        int[,] matrix = new int[5, 7];
        int[] array = new int[7];
        FillMatrix(matrix);
        FillArray(array, matrix);
        Sort(array);
        DisplayArray(array);
    }

    Ссылка: 1
    static void FillMatrix(int[,] matrix)
    {
        for (int i = 0; i < matrix.GetLength(0); i++)
            for (int j = 0; j < matrix.GetLength(1); j++)
                matrix[i, j] = new Random().Next(-100, 101);
    }

    Ссылка: 1
    static void FillArray(int[] array, int[,] matrix)
    {
        for (int i = 0; i < matrix.GetLength(1); i++)
        {
            for (int j = 0; j < matrix.GetLength(0); j++)
                array[i] += matrix[j, i];
        }
    }
}
  
```

```

Ссылка: 1
static void Sort(int[] array)
{
    int tmp;
    for(int i = 0; i < array.Length; i++)
        for(int j = 0; j < array.Length - 1; j++)
        {
            if (array[j] > array[j + 1])
            {
                tmp = array[j];
                array[j] = array[j + 1];
                array[j + 1] = tmp;
            }
        }
}

Ссылка: 1
static void DisplayArray(int[] array)
{
    for (int i = 0; i < array.Length; i++)
        Console.Write(array[i] + " ");
}
  
```

Результат роботи програми

Консоль отладки Microsoft Visual Studio

```
-189 -182 -101 -92 -26 93 107
```

Висновок

Виконуючи лабораторну роботу, я дослідив особливості алгоритмів пошуку та сортування, набув практичних навичок їх використання під час складання програмних специфікацій.