

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 № 7 з дисципліни
«Алгоритми та структури даних-1. Основи
алгоритмізації»

«Дослідження лінійного пошуку в
послідовностях»

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 7

Дослідження лінійного пошуку в послідовностях

Мета – дослідити методи послідовного пошуку у впорядкованих і неупорядкованих послідовностях та набуті практичних навичок їх використання під час складання програмних специфікацій.

Варіант 6.

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

1. Опису трьох змінних індексованого типу з 10 символьних значень.
2. Ініціювання двох змінних: 1-ий масив = $120 - i$, 2-ий масив = $110 + i$.
3. Ініціювання третьої змінної рівними значеннями двох попередніх змінних.
4. Обробки третьої змінної: знаходження кількості елементів, коди яких менше 115

1. Постановка задачі

Результатом розв'язку є кількість елементів третього масиву, коди яких менше 115

2. Побудова математичної моделі

Таблиця змінних

Змінна	Тип	Ім'я	Призначення
Перший масив	Символ	array1	Проміжне дане
Другий масив	Символ	array2	Проміжне дане
Третій масив	Символ	array3	Проміжне дане
Лічильник циклу	Натуральний	i	Проміжне дане
Лічильник вложеного циклу	Натуральний	j	Проміжне дане
Кількість елементів	Цілий	elementsAmount	Вихідні дані

Таким чином математичне формулювання задачі зводиться до обчислення двох масивів розмірністю 10, де елементи знаходяться за формулами:

1-ий масив: $120 - i$,

2-ий масив : $110 + i$,

Елементи третього масиву, це повторювальні елементи першого і другого масиву, тобто їх перетин.

3. Розв'язання

Програмні специфікації запишемо у псевдокодi та графічній формi у вигляді блок-схеми.

- Крок 1. Визначимо основні дії
- Крок 2. Деталізуємо дію оголошення змінних
- Крок 3. Деталізуємо дію заповнення масиву
- Крок 4. Деталізуємо дію отримання масиву з спільних елементів
- Крок 5. Деталізуємо дію знаходження кількості елементів масиву

Псевдокод

Крок 1

Підпрограма FillArray(array, startChar, minus = 1)

Заповнення масиву

Все підпрограма

Підпрограма GetCommonArray(array1, array2)

Отримання масиву з спільних елементів

Все підпрограма

Підпрограма GetElementsAmount(array)

Знаходження кількості елементів масиву

Все підпрограма

початок

Оголошення змінних

Виклик функції FillArray(array1, 120, -1)

Виклик функції FillArray(array2, 110)

Виклик функції GetCommonArray(array1, array2)

Виведення elementsAmount

кінець

Крок 2

Підпрограма FillArray(array, startChar, minus = 1)

Заповнення масиву

Все підпрограма

Підпрограма GetCommonArray(array1, array2)

Отримання масиву з спільних елементів

Все підпрограма

Підпрограма GetElementsAmount(array)

Знаходження кількості елементів масиву

Все підпрограма

початок

array1 = new char(10)

array2 = new char(10)

array3 = new char(10)

Виклик функції FillArray(array1, 120, -1)

Виклик функції FillArray(array2, 110)

Виклик функції GetCommonArray(array1, array2)

Виведення **elementsAmount**

кінець

Крок 3

Підпрограма FillArray(array, startChar, minus = 1)

для і від 0 до 10 повторити

array[i] := char(startChar + i * minus)

все повторити

Все підпрограма

Підпрограма GetCommonArray(array1, array2)

Отримання масиву з спільних елементів

Все підпрограма

Підпрограма GetElementsAmount(array)

Знаходження кількості елементів масиву

Все підпрограма

початок

array1 = new char(10)

array2 = new char(10)

array3 = new char(10)

Виклик функції FillArray(array1, 120, -1)

Виклик функції FillArray(array2, 110)

Виклик функції GetCommonArray(array1, array2)

Виведення **elementsAmount**

кінець

Крок 4

Підпрограма FillArray(array, startChar, minus = 1)

для і від 0 до 10 повторити

array[i] := char(startChar + i * minus)

все повторити

Все підпрограма

Підпрограма GetCommonArray(array1, array2)

для і від 0 до array1.Length повторити

для j від 0 до array2.Length повторити

якщо array1[i] == array2[j]

то

commonArray[i] = array1[i]

все повторити

все повторити

Все підпрограма

Підпрограма GetElementsAmount(array)

Знаходження кількості елементів масиву

Все підпрограма

початок

array1 = new char(10)

array2 = new char(10)

array3 = new char(10)

Виклик функції FillArray(array1, 120, -1)

Виклик функції FillArray(array2, 110)

Виклик функції GetCommonArray(array1, array2)

Виведення **elementsAmount**

кінець

Крок 4

Підпрограма FillArray(array, startChar, minus = 1)

для i від 0 до 10 повторити

array[i] := char(startChar + i * minus)

все повторити

Все підпрограма

Підпрограма GetCommonArray(array1, array2)

для i від 0 до array1.Length повторити

для j від 0 до array2.Length повторити

якщо array1[i] == array2[j]

то

commonArray[i] = array1[i]

все повторити

все повторити

Все підпрограма

Підпрограма GetElementsAmount(array)

для i від 0 до array.Length повторити

якщо array[i] < 115 && array[i] !=

то

elementsAmount := elementsAmount + 1

все повторити

Все підпрограма

початок

array1 = new char(10)

array2 = new char(10)

array3 = new char(10)

Виклик функції FillArray(array1, 120, -1)

Виклик функції FillArray(array2, 110)

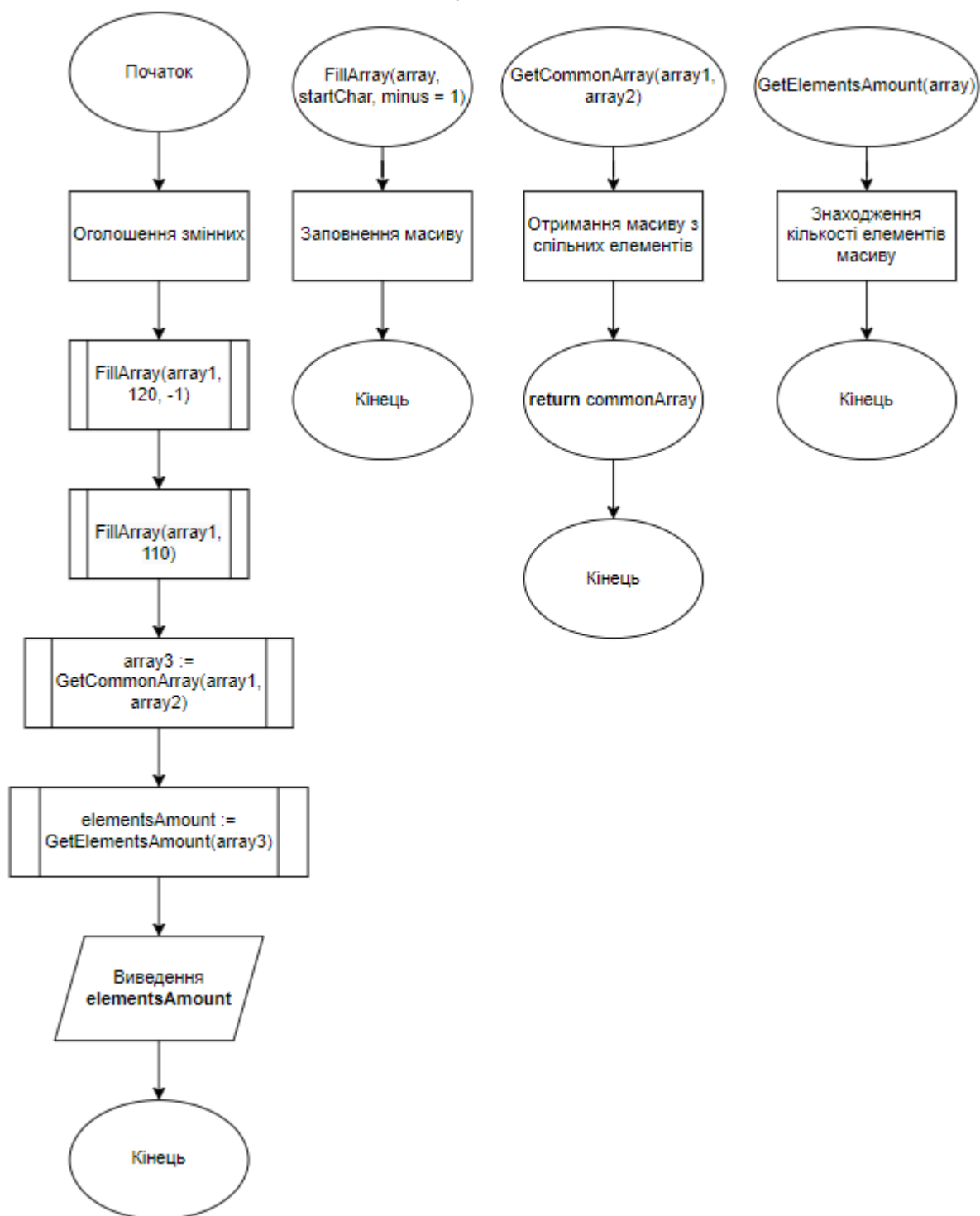
Виклик функції GetCommonArray(array1, array2)

Виведення **elementsAmount**

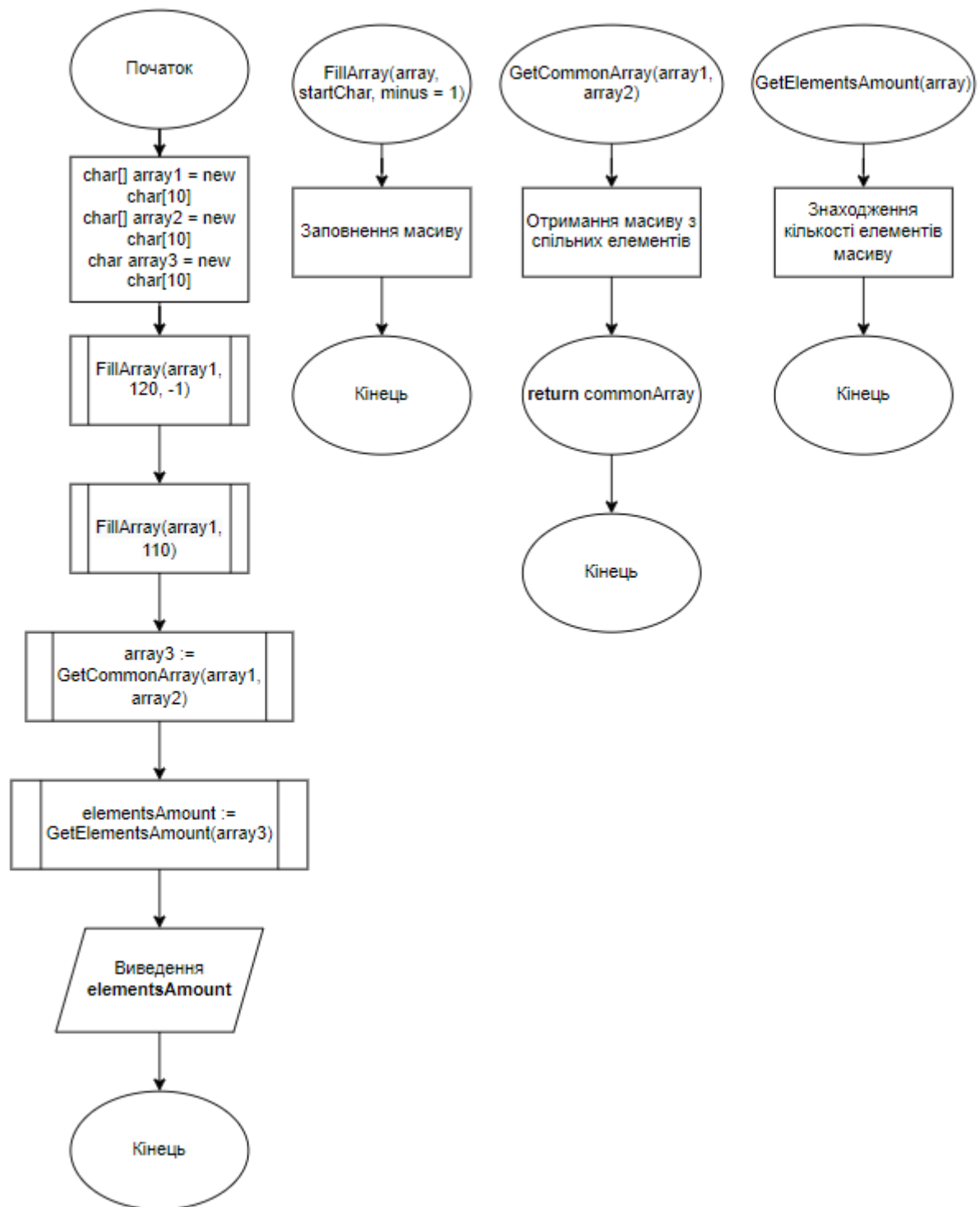
кінець

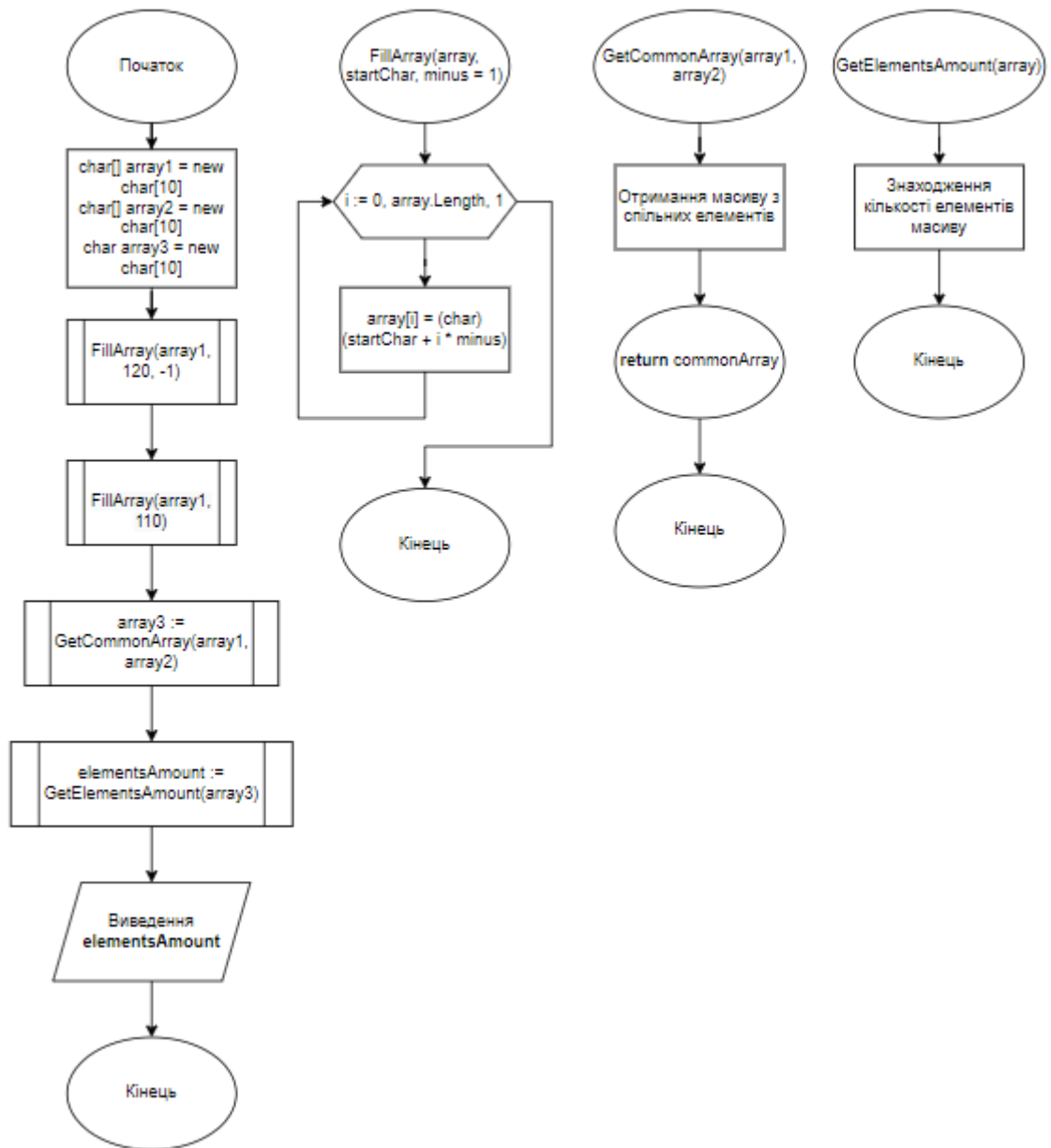
Блок-схема

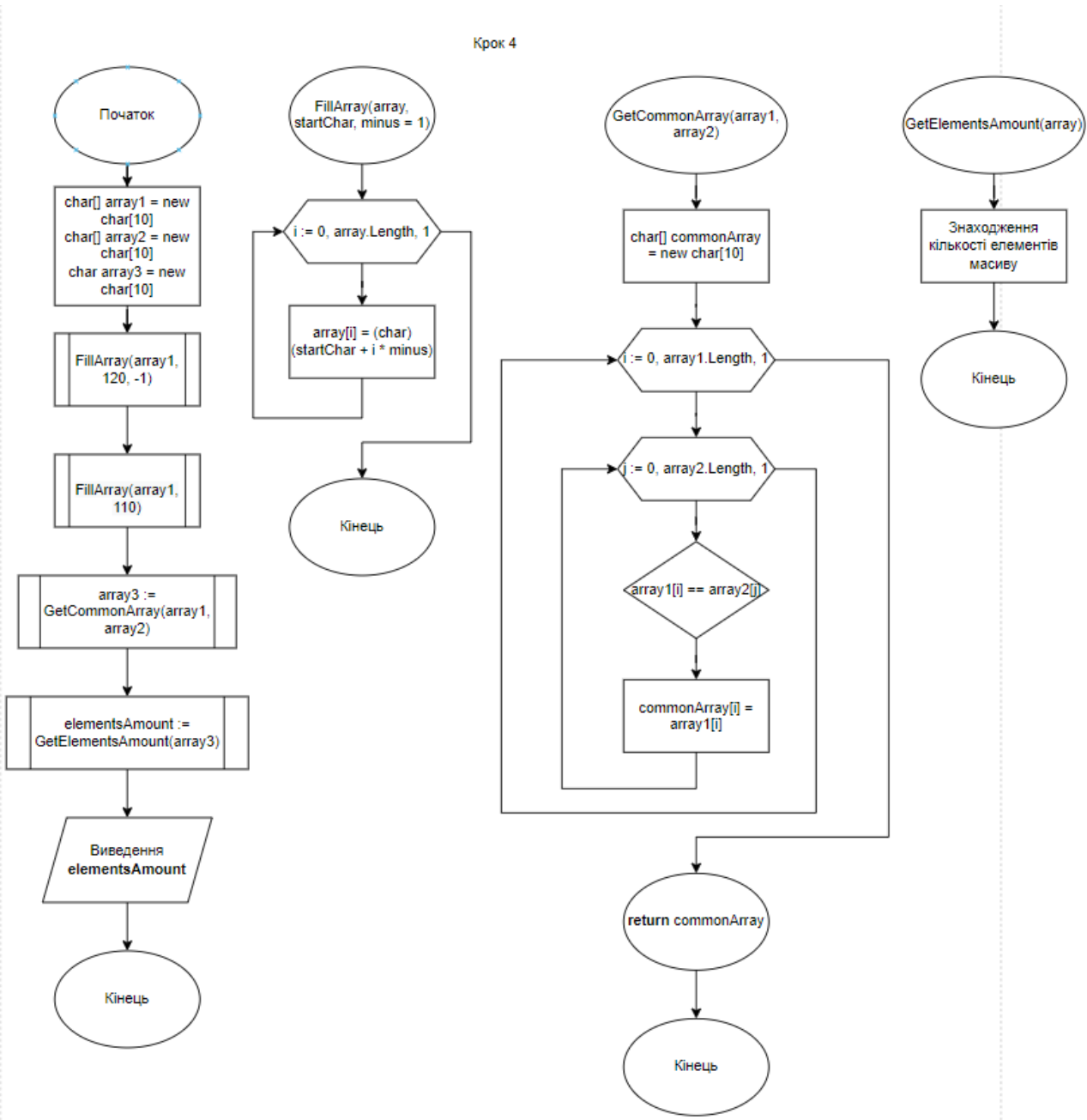
Крок 1

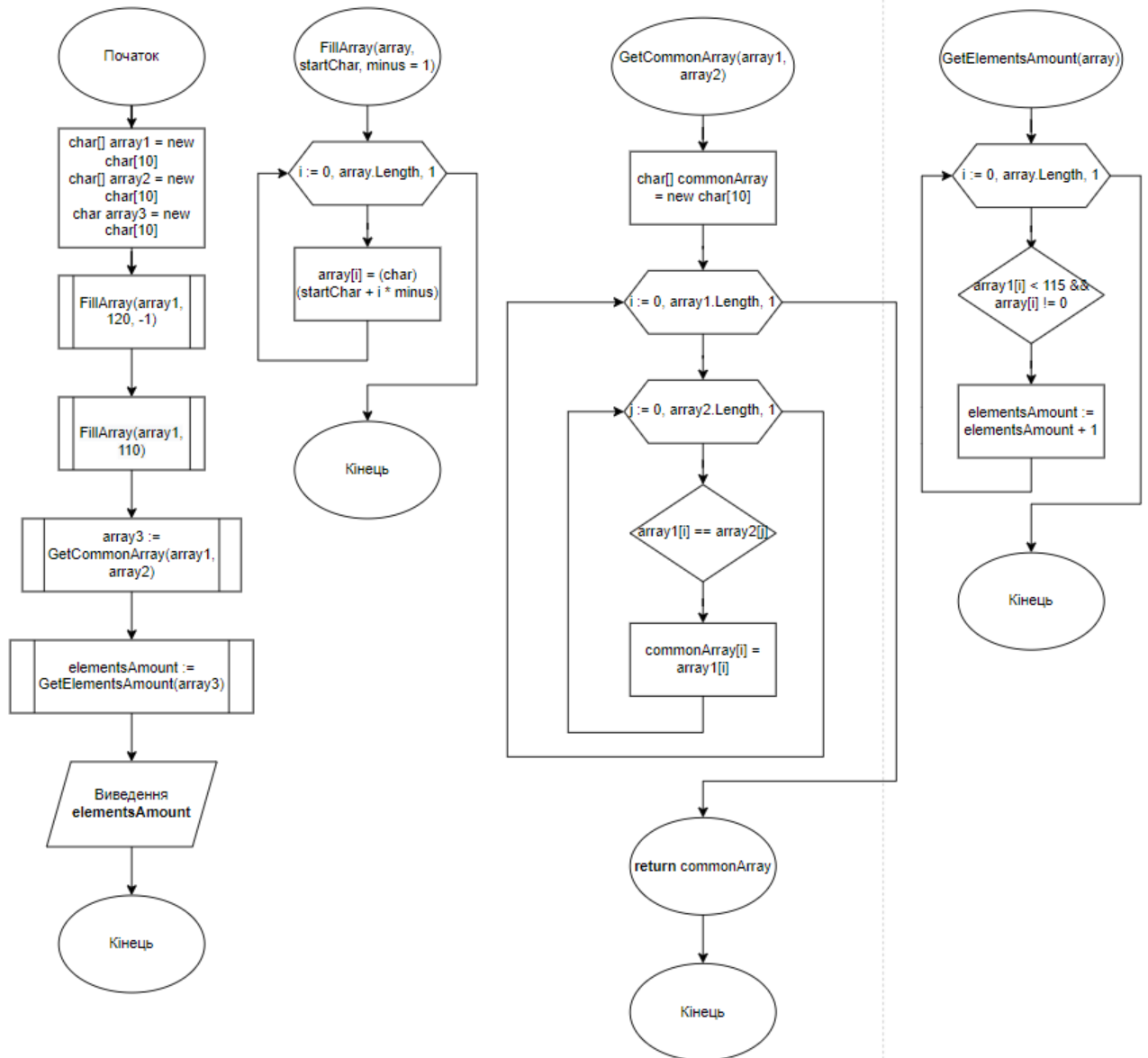


Крок 2









Код програми

```

1  using System;
2
3  namespace ASDLab7
4  {
5      Ссылка: 0
6      class Program
7      {
8          Ссылка: 0
9          static void Main(string[] args)
10         {
11             char[] array1 = new char[10];
12             char[] array2 = new char[10];
13             char[] array3 = new char[10];
14
15             FillArray(array1, 120, -1);
16             FillArray(array2, 110);
17
18             array3 = GetCommonArray(array1, array2);
19             Console.WriteLine(GetElementsAmount(array3));
20         }
21
22         Ссылка: 2
23         static void FillArray(char[] array, int startChar, int minus = 1)
24         {
25             for(int i = 0; i < 10; i++)
26             {
27                 array[i] = (char)(startChar + i * minus);
28             }
29         }
30
31         ссылка: 1
32         static char[] GetCommonArray(char[] array1, char[] array2)
33         {
34             char[] commonArray = new char[10];
35
36             for(int i = 0; i < array1.Length; i++)
37             {
38                 for(int j = 0; j < array2.Length; j++)
39

```

```

34             for(int j = 0; j < array2.Length; j++)
35             {
36                 if (array1[i] == array2[j])
37                     commonArray[i] = array1[i];
38             }
39             return commonArray;
40         }
41     }
42
43     ссылка: 1
44     static int GetElementsAmount(char[] array)
45     {
46         int elementsAmount = 0;
47         for(int i = 0; i < array.Length; i++)
48         {
49             if (array[i] < 115 && array[i] != 0)
50                 elementsAmount++;
51         }
52         return elementsAmount;
53     }
54 }
55

```

Висновок

Виконуючи лабораторну роботу, я дослідив особливості методів послідовного пошуку у впорядкованих та неупорядкованих послідовностях та набув практичних навичок їх використання під час складання програмних специфікацій.