

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 1

Виконав студент ІП-15, Багачок Вадим Дмитрович

(шифр, прізвище, ім'я, по батькові)

Перевірив - Вечерковська А.С

(прізвище, ім'я, по батькові)

Київ 2021

Лабораторна робота 7
Дослідження лінійного пошуку в послідовностях

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

B-1

варіанти завдань

Таблиця

№	Вираз для обчислення елемента		Знайти
	1-го масиву	2-го масиву	
1	117 + i	127 – 2 * i	Суму двох мінімальних елементів

Постановка задачі:

Спочатку потрібно описати три змінні індексованого типу з 10 символічних значень. На другому кроці потрібно заповнити перші дві змінні за допомогою формул. першу змінну заповніть за допомогою формули (1). Другу змінну заповніть за допомогою формули (2) . На третьому кроці заповнюємо третій масив однаковими змінними з перших двох масивів. На останньому четвертому кроці потрібно знайти два мінімальні елементи третьої змінної і скласти їх.

Побудова математичної моделі:

Спочатку створюємо змінні індексованого типу за допомогою конструкції оголошення масиву .Потім перші дві змінні заповніть за допомогою циклу , і формул які дані в таблиці (117+i та 127-2*i) . На третьому кроці за допомогою циклу for заповніть третю змінну рівними елементами з перших двох змінних. На четвертому кроці пишемо метод, який за допомогою алгоритму сортування вибіркою знаходитиме мінімальні мінімальні елементи третього масиву. Закінчуємо сортування після того, як знайдемо перші два мінімальні елементи відповідно до умови завдання. Після того як ми знайшли мінімальні елементи масиву, складаємо їх і результат повертаємо користувачеві.

Змінна	Тип	Ім'я	Призначення
array1	ціле	array1	Проміжні дані
array2	ціле	array2	Проміжні дані
array3	ціле	array3	Проміжні дані
i	ціле	m	Проміжні дані
j	ціле	j	Проміжні дані
index	ціле	index	Проміжні дані
min	ціле	min	Проміжні дані
result	ціле	result	Проміжні дані
sum	ціле	sum	Кінцеві дані

tmp	ціле	tmp	Проміжні дані
-----	------	-----	---------------

Псевдокод :

Підпрограма

```
Sampling( array3)
Повторити для i від 0 до 2
    min = i + 1
        повторити для j від min до array3.length
            якщо array3[min]>array3[i] і array3[min]>0 І array3[j]>0
                то
                    min=j
            все якщо
            якщо array3[i]>array3[min]
        то
            tmp =array3[i]
            array3[i]=array3[min]
            array3[min]=tmp
        все якщо
        все повторити
    все повторити
    result=array3[0]+array3[1]
return result
```

Крок1

```
Початок
[] array1-[10]
[] array1-[10]
[] array1-[10]
```

Кінець

Крок 2

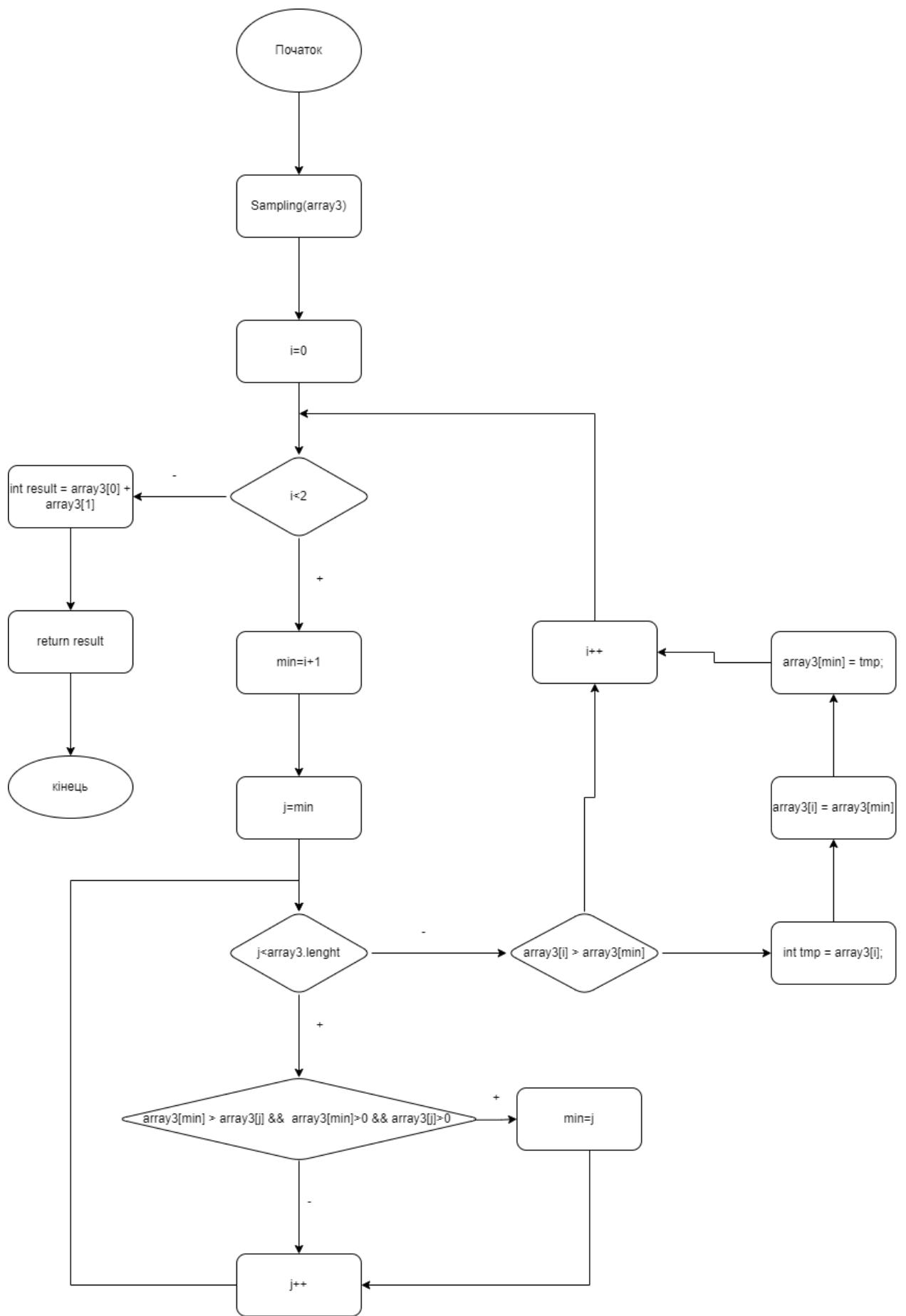
```
Початок
    повторити для j від 0 до array1.length І array2.length
        array1[i]=117+i
        array2[i]=127-2*i
    все повторити
Кінець
```

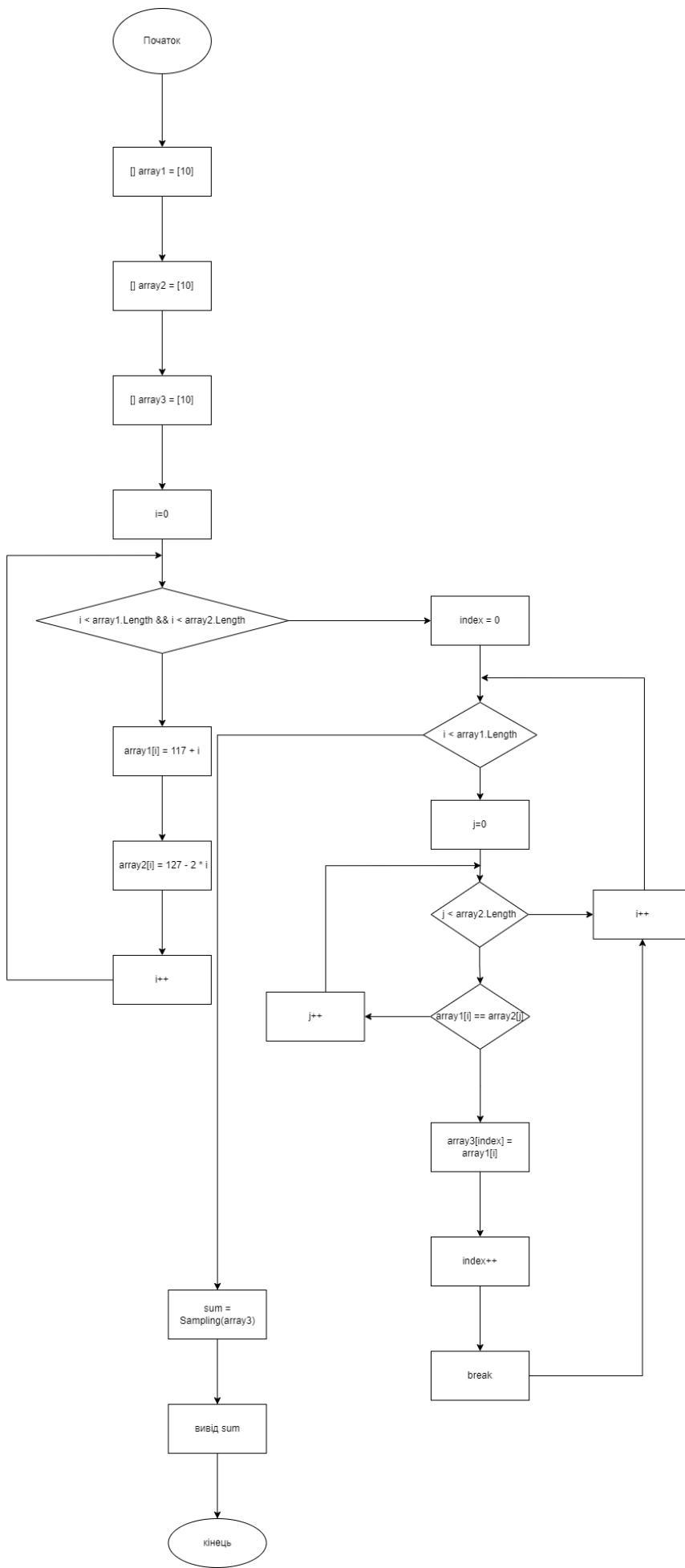
Крок 3

```
Початок
    index=0
    повторити для j від 0 до length
        якщо array1[i]==array2[j]
            то array3[index]=array1[i]
            index++
            break
    все якщо
    все повторити
```

Крок 4

```
Почток
sum=Sampling(array3)
виведення sum
Кінець
```





```
1  using System;
2
3  namespace ASD_7.lab
4  {
5      class Program
6      {
7          static void Main(string[] args)
8          {
9              int[] array1 = new int[10];
10             int[] array2 = new int[10];
11             int[] array3 = new int[10];
12             for (int i = 0; i < array1.Length && i < array2.Length; i++)
13             {
14                 array1[i] = 117 + i;
15                 array2[i] = 127 - 2 * i;
16             }
17             int index = 0;
18             for (int i = 0; i < array1.Length; i++)
19             {
20                 for (int j = 0; j < array2.Length; j++)
21                 {
22                     if (array1[i] == array2[j])
23                     {
24                         array3[index] = array1[i];
25                         index++;
26                         break;
27                     }
28                 }
29             }
30             int sum = Sampling(array3);
31             Console.WriteLine(sum);
32             Console.ReadKey();
33         }
34     }
35
36     private static int Sampling(int[] array3)
37     {
38         for (int i = 0; i < 2; i++)
39         {
40             int min = i + 1;
41             for (int j = min; j < array3.Length; j++)
42             {
43                 if (array3[min] > array3[j] && array3[min]>0 &&array3[j]>0 )
44                 {
45                     min = j;
46                 }
47             }
48             if (array3[i] > array3[min])
49             {
50                 int tmp = array3[i];
51                 array3[i] = array3[min];
52                 array3[min] = tmp;
53             }
54         }
55         int result = array3[0] + array3[1];
56         return result;
57     }
58 }
```

```
int[] array1 = new int[10];
int[] array2 = new int[10];
int[] array3 = new int[10];
for (int i = 0; i < array1.Length && i < array2.Length; i++)
{
    array1[i] = 117 + i;
    array2[i] = 127 - 2 * i;
}
int index = 0;
for (int i = 0; i < array1.Length; i++)
{
    for (int j = 0; j < array2.Length; j++)
    {
        if (array1[i] == array2[j])
        {
            array3[index] = array1[i];
            index++;
            break;
        }
    }
}
int sum = Sampling(array3);
Console.WriteLine("Сума двох мінімальних елементів = " + sum);
Console.ReadKey();
}

private static int Sampling(int[] array3)
{
    for (int i = 0; i < 2; i++)
    {
        int min = i + 1;
        for (int j = min; j < array3.Length; j++)
        {
            if (array3[min] > array3[j] && array3[min]>0 && array3[j]>0 )
            {
                min = j;
            }
        }

        if (array3[i] > array3[min])
        {
            int tmp = array3[i];
            array3[i] = array3[min];
            array3[min] = tmp;
        }
    }
    int result = array3[0] + array3[1];
    return result;
}
```

Висновки: Ми дослідили методи послідовного пошуку у впорядкованих і невпорядкованих послідовностях та набули практичних навичок їх використання під час складання програмних специфікацій.