

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 22

Виконав студент ІП-13, Музичук Віталій Андрійович  
(шифр, прізвище, ім'я, по батькові)

Перевірила Вєчерковська Анастасія Сергіївна  
(прізвище, ім'я, по батькові)

## Лабораторна робота 6

### Дослідження лінійного пошуку в послідовностях

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

### Варіант 22

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

1. Опису трьох змінних індексованого типу з 10 символних значень.
2. Ініціювання двох змінних виразами згідно з варіантом (табл. 1).
3. Ініціювання третьої змінної рівними значеннями двох попередніх змінних.
4. Обробки третьої змінної згідно з варіантом.

22	$92 - 3 * i$	$71 + 3 * i$	Середнє арифметичне елементів, коди яких менше 82
----	--------------	--------------	---

### Постановка задачі

Розбиваємо кожен підпункт завдання на окремі підпрограми. Підпрограми `initArray1`, `initArray2`, `initArray3` будуть ініціалізовувати три масиви за заданими даними. Далі `averageSum` буде обробляти масив згідно з варіантом за допомогою ітераційних циклів. Також нам знадобиться підпрограма яка буде виводити масив на екран для наочності. Результатом виконання програми є середнє арифметичне елементів коди яких менші за 82.

### Побудова математичної моделі

Складемо таблицю змінних

Змінна	Тип	Ім'я	Призначення
Перший масив	Символьний	<i>array1</i>	Проміжне значення
Другий масив	Символьний	<i>array2</i>	Проміжне значення

Третій масив	Символьний	<i>array3</i>	Проміжне значення
Підпрограма для ініціалізації першого масиву	void	<i>initArray1</i>	Ініціалізація першого масиву
Підпрограма для ініціалізації другого масиву	void	<i>initArray2</i>	Ініціалізація другого масиву
Підпрограма для ініціалізації третього масиву	void	<i>initArray3</i>	Ініціалізація третього масиву
Підпрограма для знаходження результату	Дійсний	<i>averageSum</i>	Середнє арифметичне елементів коди яких менші за 82
Підпрограма для виведення масиву	void	<i>arrayOut</i>	Виведення масиву в консоль
Результат	Дійсний	<i>result</i>	Кінцеве значення
Сума елементів масиву коди яких менше 82	Цілий	<i>sum</i>	Проміжне значення
Кількість елементів масиву коди яких менше 82	Цілий	<i>k</i>	Проміжне значення

**InitArray1** – підпрограма яка заповнює масив через ітераційний цикл за формулою  $92 - 3 * i$ .

**initArray2** - підпрограма яка заповнює масив через ітераційний цикл за формулою  $71 + 3 * i$ .

**initArray3** - підпрограма яка заповнює масив через ітераційний цикл спільними елементами попередніх масивів.

**averageSum** – підпрограма, яка за допомогою ітераційного циклу перевіряє кожен елемент чи його код менше за 82, і якщо це так додає його до суми. Також змінна *k* рахує кількість доданих елементів, що потім порахувати середнє арифметичне.

**arrayOut** – підпрограма, яка поелементно виводить в консоль дані з масиву.

## Розв'язання

Програмні специфікації запишемо у псевдокоді та графічній формі у вигляді блок-схеми.

*Крок 1.* Визначаємо основні дії

*Крок 2.* Створення масивів

*Крок 3.* Ініціалізація першого масиву

*Крок 4.* Ініціалізація другого масиву

*Крок 5.* Ініціалізація третього масиву

*Крок 6.* Обробка третього масиву й виведення результату

## Псевдокод

*крок 1*

**початок**

створення масивів

ініціалізація першого масиву

ініціалізація другого масиву

ініціалізація третього масиву

обробка третього масиву й виведення результату

**кінець**

*крок 2*

**початок**

array1[10]

array2[10]

array3[10]

ініціалізація першого масиву

ініціалізація другого масиву

ініціалізація третього масиву

обробка третього масиву й виведення результату

**кінець**

*крок 3*

**початок**

array1[10]

array2[10]

array3[10]

initArray1(array1)

arrayOut(array1)

ініціалізація другого масиву

ініціалізація третього масиву

обробка третього масиву й виведення результату

**кінець**

*крок 4*

**початок**

array1[10]

array2[10]

array3[10]

initArray1(array1)

arrayOut(array1)

initArray2(array2)

arrayOut(array2)

ініціалізація третього масиву

обробка третього масиву й виведення результату

**кінець**

*крок 5*

**початок**

array1[10]

array2[10]

array3[10]

initArray1(array1)

arrayOut(array1)

initArray2(array2)

arrayOut(array2)

initArray3(array3)

arrayOut(array3)

обробка третього масиву й виведення результату

**кінець**

*крок 6*

**початок**

array1[10]

array2[10]

array3[10]

initArray1(array1)

arrayOut(array1)

initArray2(array2)

arrayOut(array2)

initArray3(array1, array2, array3)

arrayOut(array3)

result = averageSum(array3)

**виведення result**

**кінець**

**Псевдокод підпрограм:**

**arrayOut(array)**

для  $i$  від 0 до 9 повторити

    виведення array[i]

    виведення ' '

все повторити

виведення '\n'

**initArray1(array1)**

для  $i$  від 0 до 9 повторити

$\text{array1}[i] = 92 - 3 * i$

все повторити

**initArray2(array2)**

для  $i$  від 0 до 9 повторити

$\text{array1}[i] = 71 + 3 * i$

все повторити

**initArray3(array1, array2, array3)**

для  $j$  від 0 до 9 повторити

для  $i$  від 0 до 9 повторити

якщо  $\text{array1}[j] == \text{array2}[i]$

$\text{array3}[j] = \text{array1}[j]$

все повторити

все повторити

**averageSum(array3)**

$k = 0$

$\text{sum} = 0$

для  $i$  від 0 до 9 повторити

$\text{ifTrue} = \text{array3}[i] > 0$

якщо  $\text{array3}[i] < 82$  і  $\text{ifTrue}$

$\text{sum} = \text{sum} + \text{array}[i]$

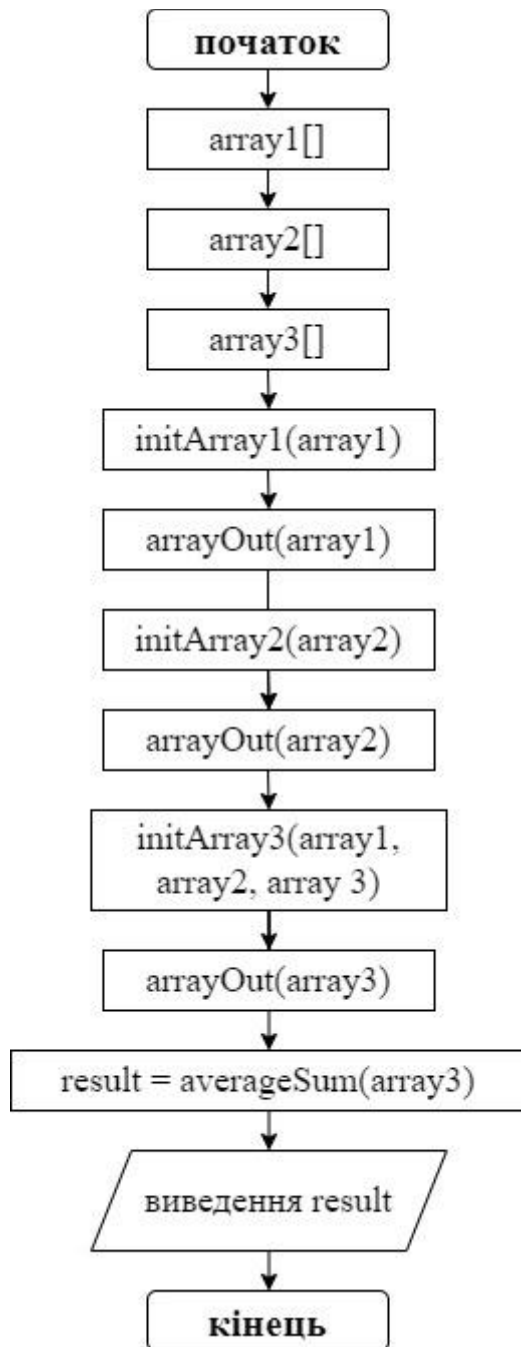
$k = k + 1$

все повторити

$\text{result} = \text{sum} / k$

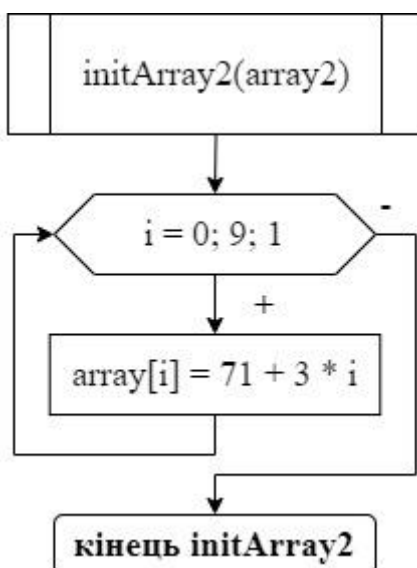
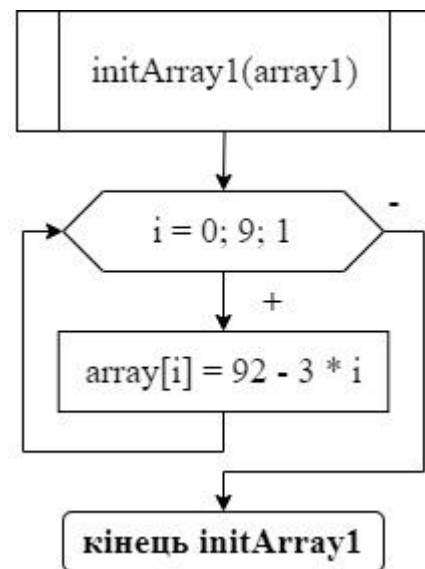
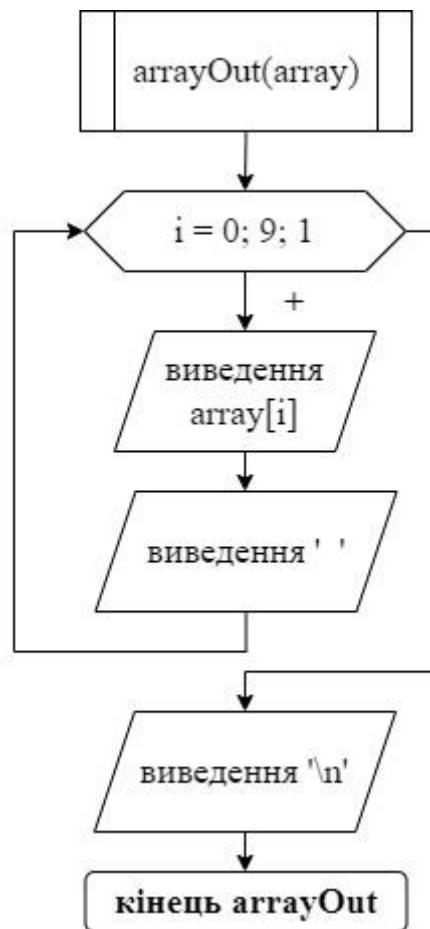
повернути  $\text{result}$

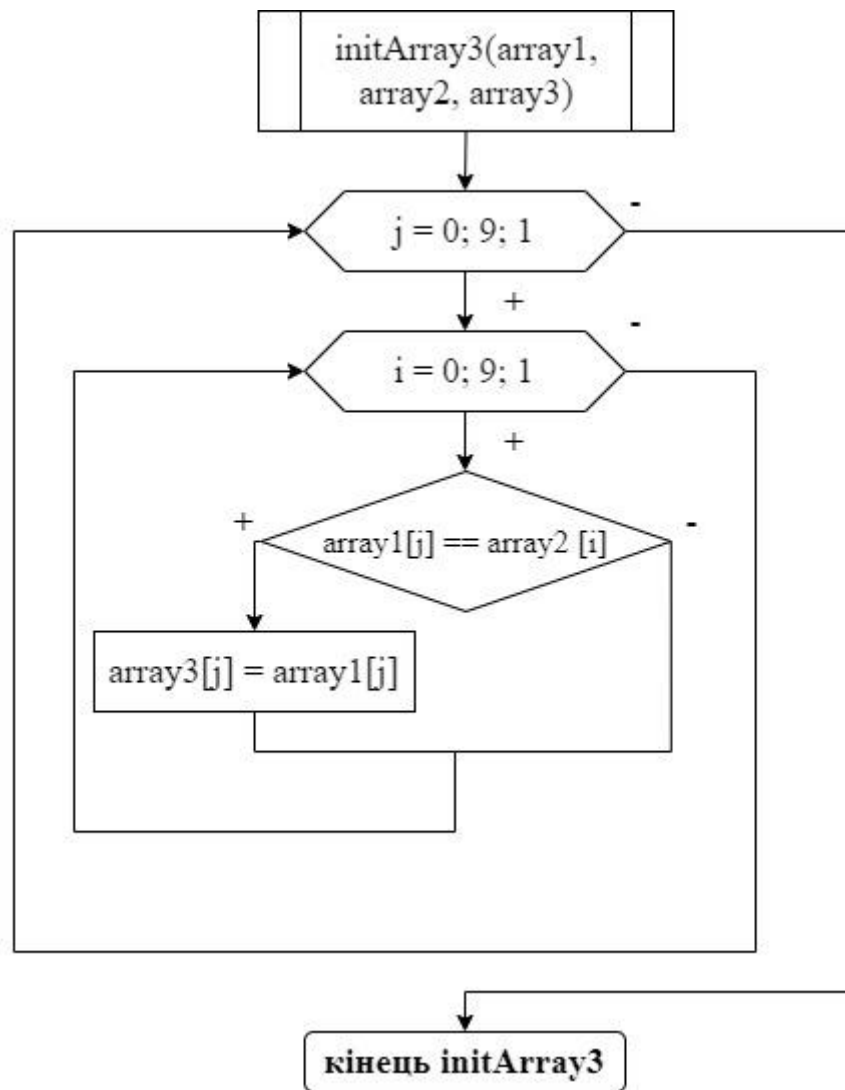
## Блок-схема

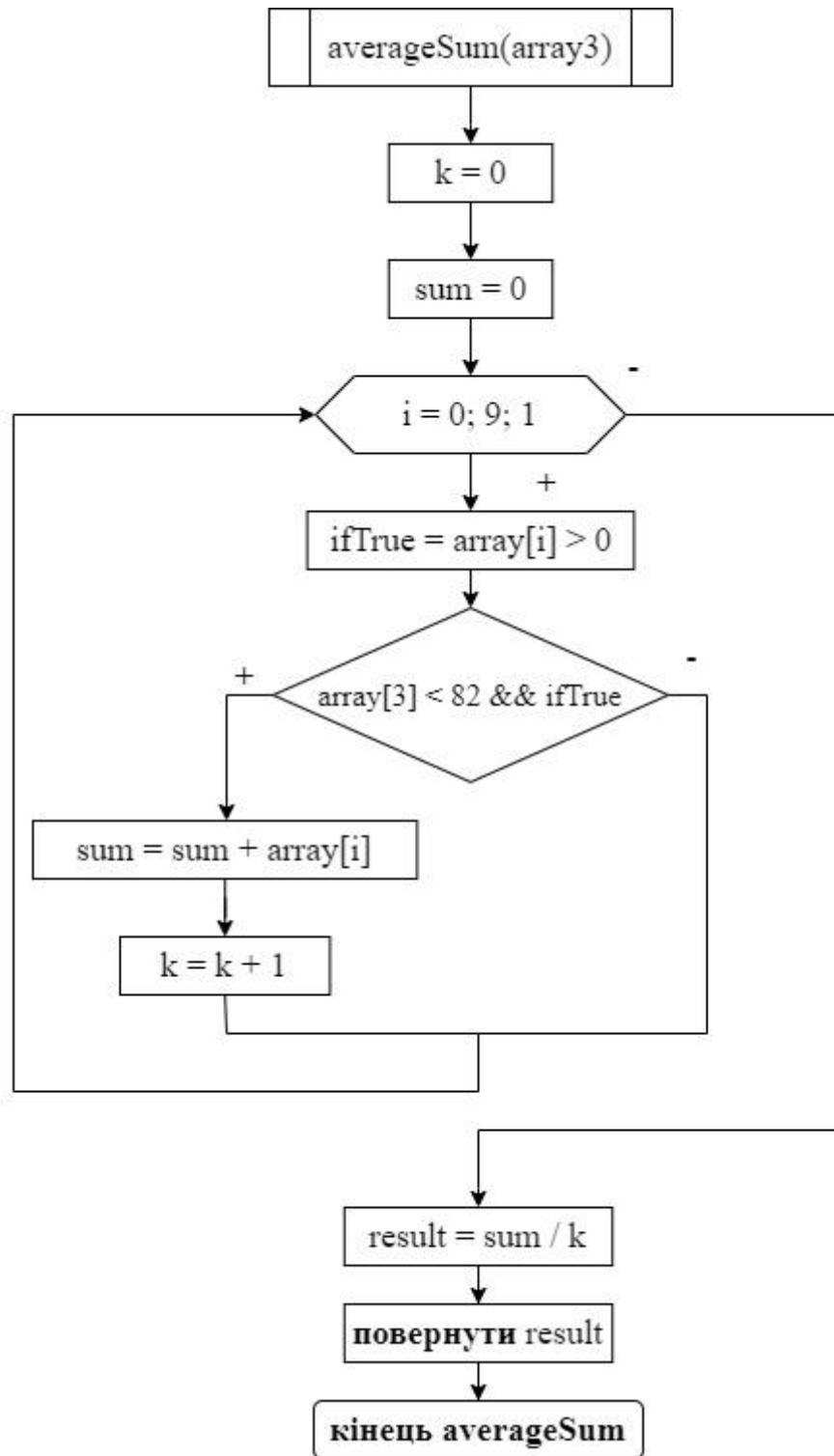




Блок схеми підпрограм:







Код програми:

```
#include <iostream>
using namespace std;

void arrayOut(char*);
void initArray1(char*);
void initArray2(char*);
void initArray3(char*, char*, char*);
float averageSum(char*);

int main() {
    char array1[10] = {}; char array2[10] = {};
    char array3[10] = {};

    initArray1(array1);
    cout << "First array: "; arrayOut(array1);

    initArray2(array2);
    cout << "Second array: "; arrayOut(&array2[0]);

    initArray3(array1, array2, array3);
    cout << "Third array: "; arrayOut(array3);

    float result = averageSum(array3);
    cout << "Average sum of elements: " << result << endl;
}
```

```
void arrayOut(char* array)
{
    for (int i = 0; i < 10; i++)
    {
        cout << *(array + i) << " ";
    }
    cout << '\n';
}

void initArray1(char* array1)
{
    for (int i = 0; i < 10; i++)
    {
        *(array1 + i) = 92 - 3 * i;
    }
}

void initArray2(char* array2)
{
    for (int i = 0; i < 10; i++)
    {
        *(array2 + i) = 71 + 3 * i;
    }
}

void initArray3(char* array1, char* array2, char* array3)
{
    for (int j = 0; j < 10; j++)
    {
        for (int i = 0; i < 10; i++)
        {
            if (*(array1 + j) == *(array2 + i))
                *(array3 + j) = *(array1 + j);
        }
    }
}
```

```
float averageSum(char* array3)
{
    int k = 0;
    float sum = 0;
    for (int i = 0; i < 10; i++)
    {
        int arrayElement = static_cast<int>(*(array3 + i));
        if (arrayElement < 82 && arrayElement > 0)
        {
            sum += arrayElement;
            k++;
        }
    }

    return sum / k;
}
```



### Microsoft Visual Studio Debug Console

```
First array:  \ Y V S P M J G D A
Second array: G J M P S V Y \ _ b
Third array:  \ Y V S P M J G
Average sum of elements: 75.5
```

## Випробування алгоритму:

Блок	Дія
	Початок
1	Array1[10]
2	Array2[10]
3	Array3[10]
4	<b>initArray1(array1)</b>
5	Для і від 1 до 9
6	Array[1] = 71
7	Array[2] = 74
8	Array[3] = 77
9	Array[4] = 80
10	Array[...] = 83
11	Array[10] = 98
12	<b>initArray2(array2)</b>
13	Для і від 1 до 9
14	Array[1] = 98
15	Array[2] = 89
16	Array[3] = 86
17	Array[4] = 83
18	Array[...] = ...
19	Array[10] = 65
20	<b>initArray2(array2)</b>
21	Для і від 1 до 9
22	Для j від 1 до 9
23	Array3[1] = 92
24	Array3[2] = 89
25	Array3[3] = 86
26	Array3[4] = 83
27	Array3[...] = ...
28	Array3[10] = 0
29	<b>averageSum(array3)</b>
30	Для і від 1 до 9
31	Якщо array3[i] < 82 True
32	Sum = sum + array3[i]
33	K = k + 1
38	<b>Виведення result</b>
	<b>Кінець</b>

**Висновки:**

На цій практичній ми дослідили методи послідовного пошуку у впорядкованих і неупорядкованих послідовностях та набули практичних навичок їх використання під час складання програмних специфікацій. Також зробили постановку задачі склали матмодель написали псевдокод та намлювали блок-схему. Отримали очікуваний результат.