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)

Київ 2021

**Лабораторна робота 6  
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.

**Варіант 22**

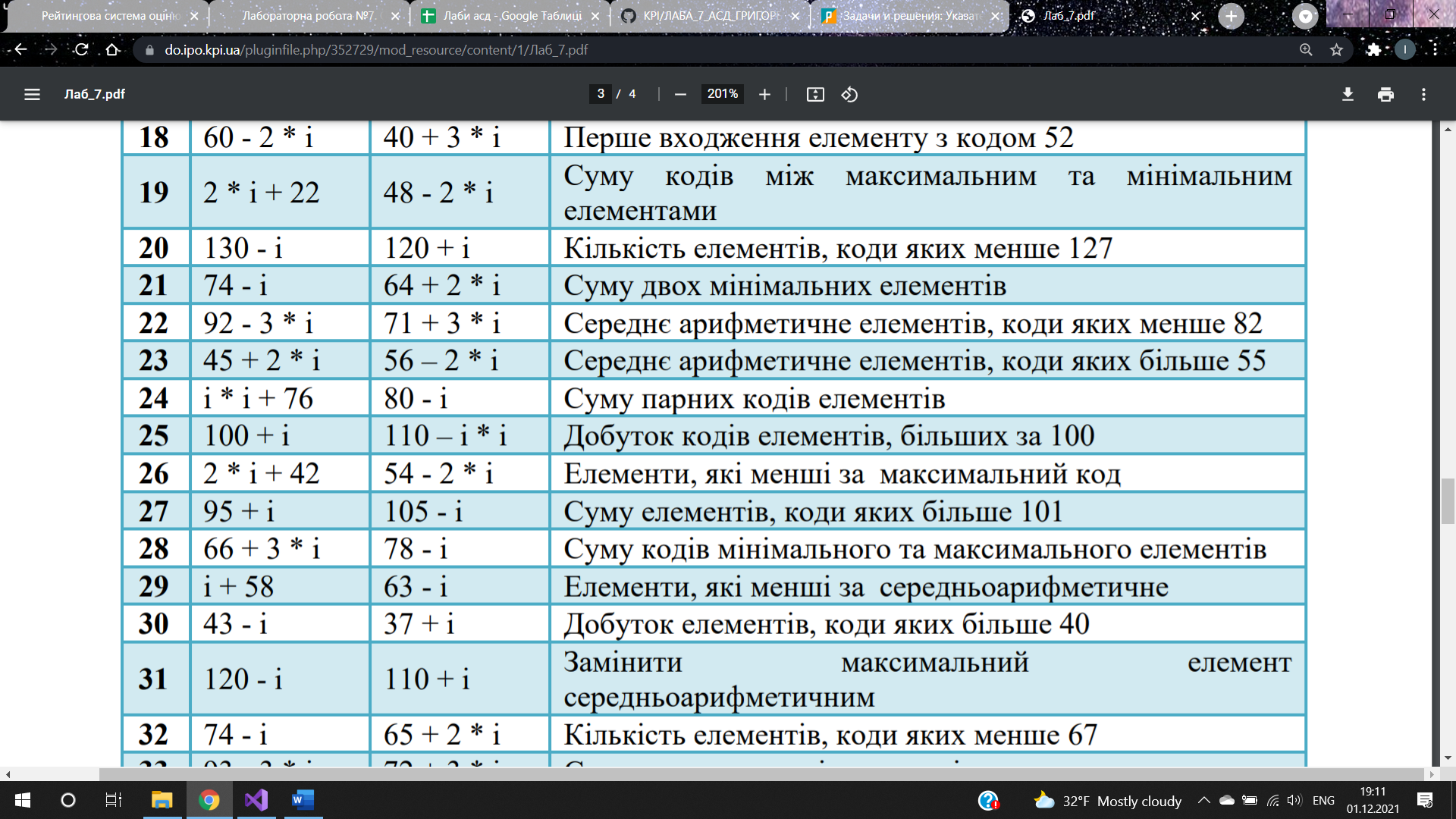
Розробити алгоритм та написати програму, яка складається з наступних дій:

1. Опису трьох змінних індексованого типу з 10 символьних 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.



**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 | *array1* | Проміжне значення |
| Другий масив | Символьний | *array2* | Проміжне значення |
| Третій масив | Символьний | *array3* | Проміжне значення |
| Підпрограма для ініціалізації першого масиву | void | *initArray1* | Ініціалізація першого масиву |
| Підпрограма для ініціалізації другого масиву | void | *initArray2* | Ініціалізація другого масиву |
| Підпрограма для ініціалізації третього масиву | void | *initArray3* | Ініціалізація третього масиву |
| Підпрограма для знаходження результату | Дійсний | *averageSum* | Середнє арифметичне елементів коди яких менші за 82 |
| Підпрограма для виведення масиву | void | *arrayOut* | Виведення масиву в консоль |
| Результат | Дійсний | *result* | Кінцеве значення |
| Сума елементів масиву коди яких менше 82 | Цілий | *sum* | Проміжне значення |
| Кількість елементів масиву коди яких менше 82 | Цілий | *k* | Проміжне значення |

**InitArray1 –** підпрограма яка заповнює масив через ітераційний цикл за формулою 92 – 3\* і.

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

**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 і від 0 до 9 повторити**

**виведення** array[i]  
 **виведення ‘ ’  
 все повторити  
 виведення ‘\n’**

**initArray1(array1)**

**для і від 0 до 9 повторити**

array1[i] = 92 – 3 \* i

**все повторити**

**initArray2(array2)**

**для і від 0 до 9 повторити**

array1[i] = 71 + 3 \* i

**все повторити**

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

**для j від 0 до 9 повторити**

**для і від 0 до 9 повторити**

**якщо** array1[j] == array2[i]

array3[j] = array1[j]

**все повторити**

**все повторити**

**averageSum(array3)**

k = 0  
 sum = 0  
 **для і від 0 до 9 повторити**

ifTrue = array3[i] > 0

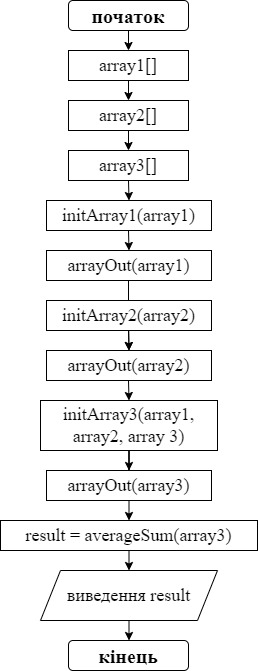
**якщо** array3[i] < 82 **і** ifTrue

sum = sum + array[i]  
 k = k + 1

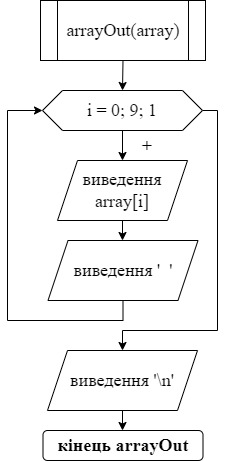
**все повторити** result = sum / k

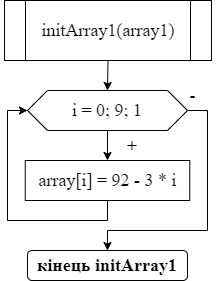
**повернути** result

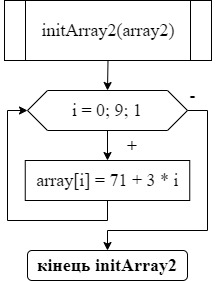
**Блок-схема**

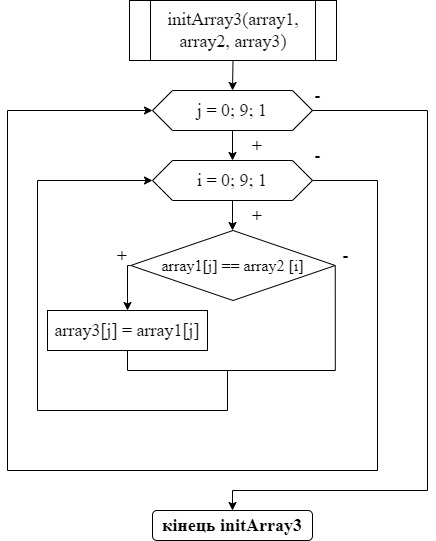


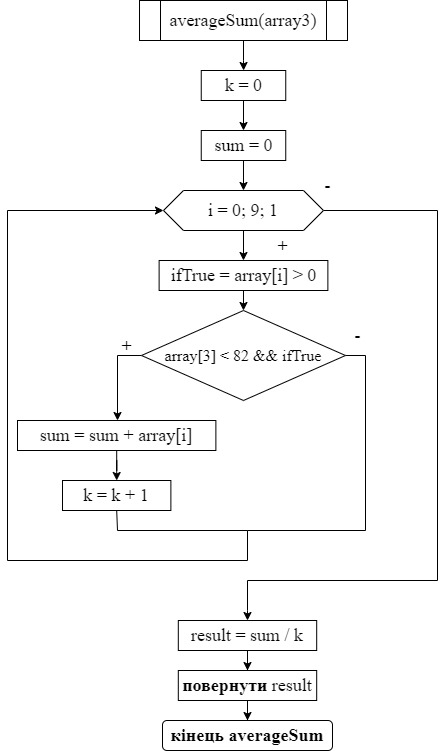
*Блок схеми підпрограм:*



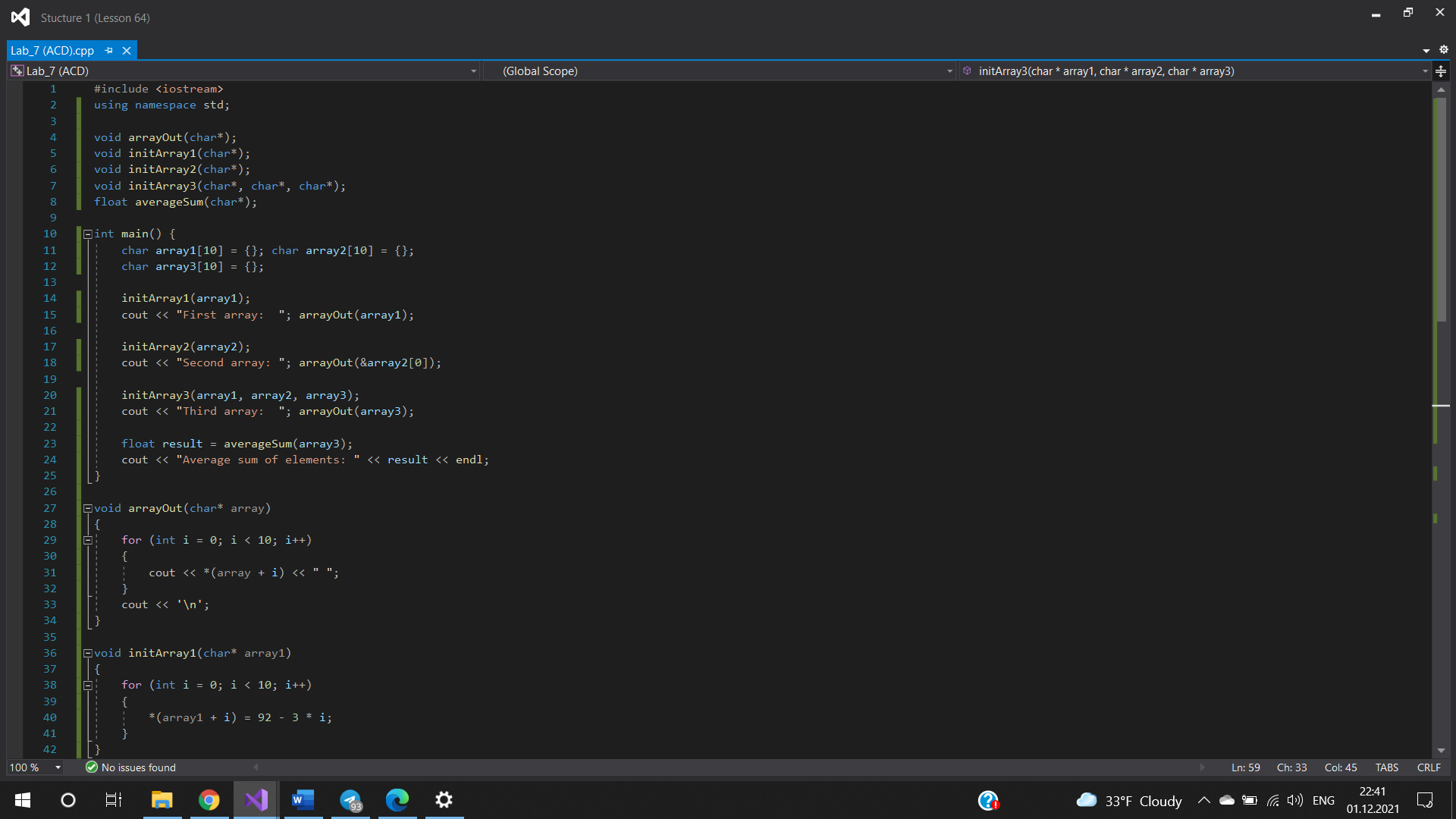


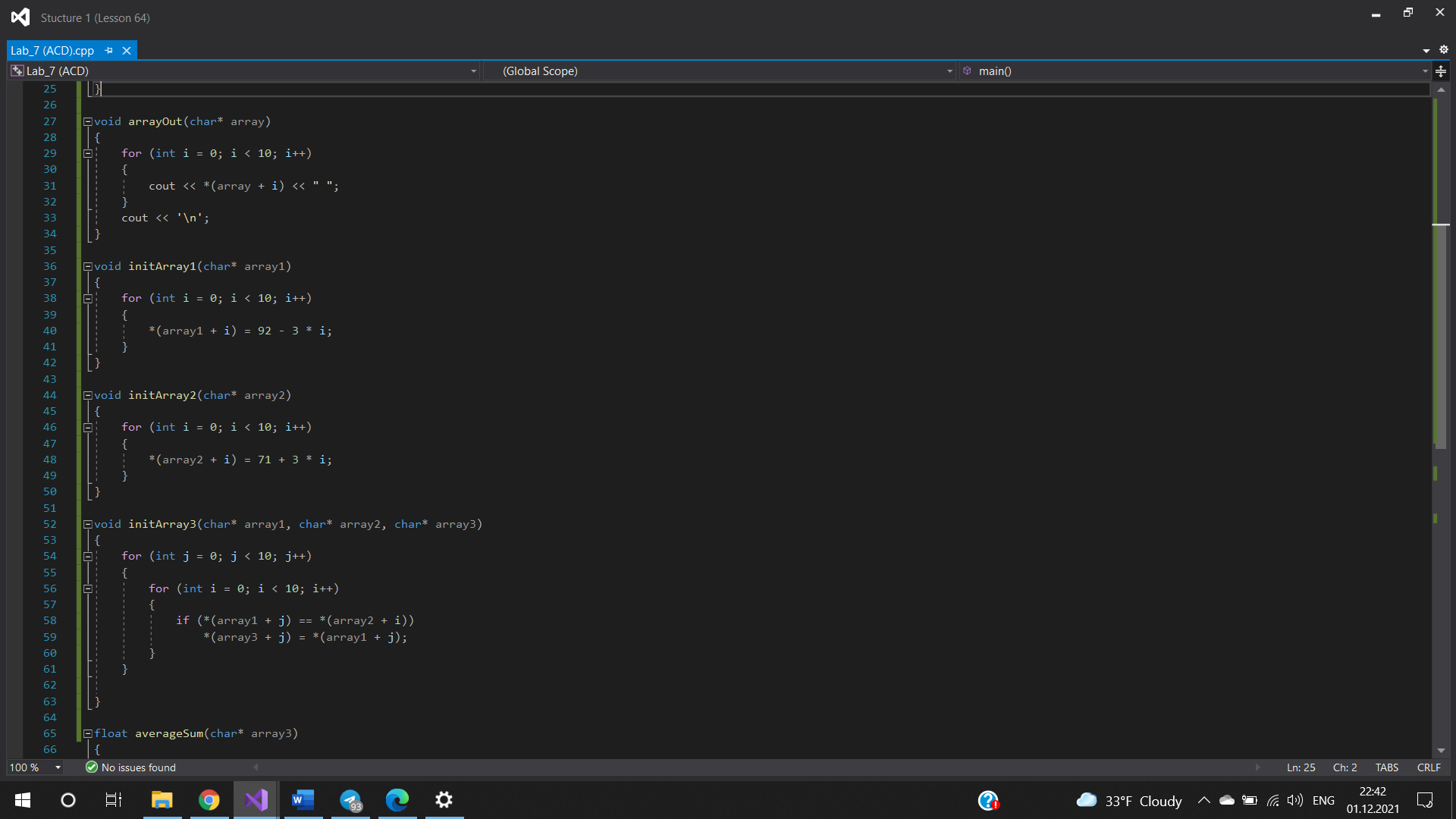


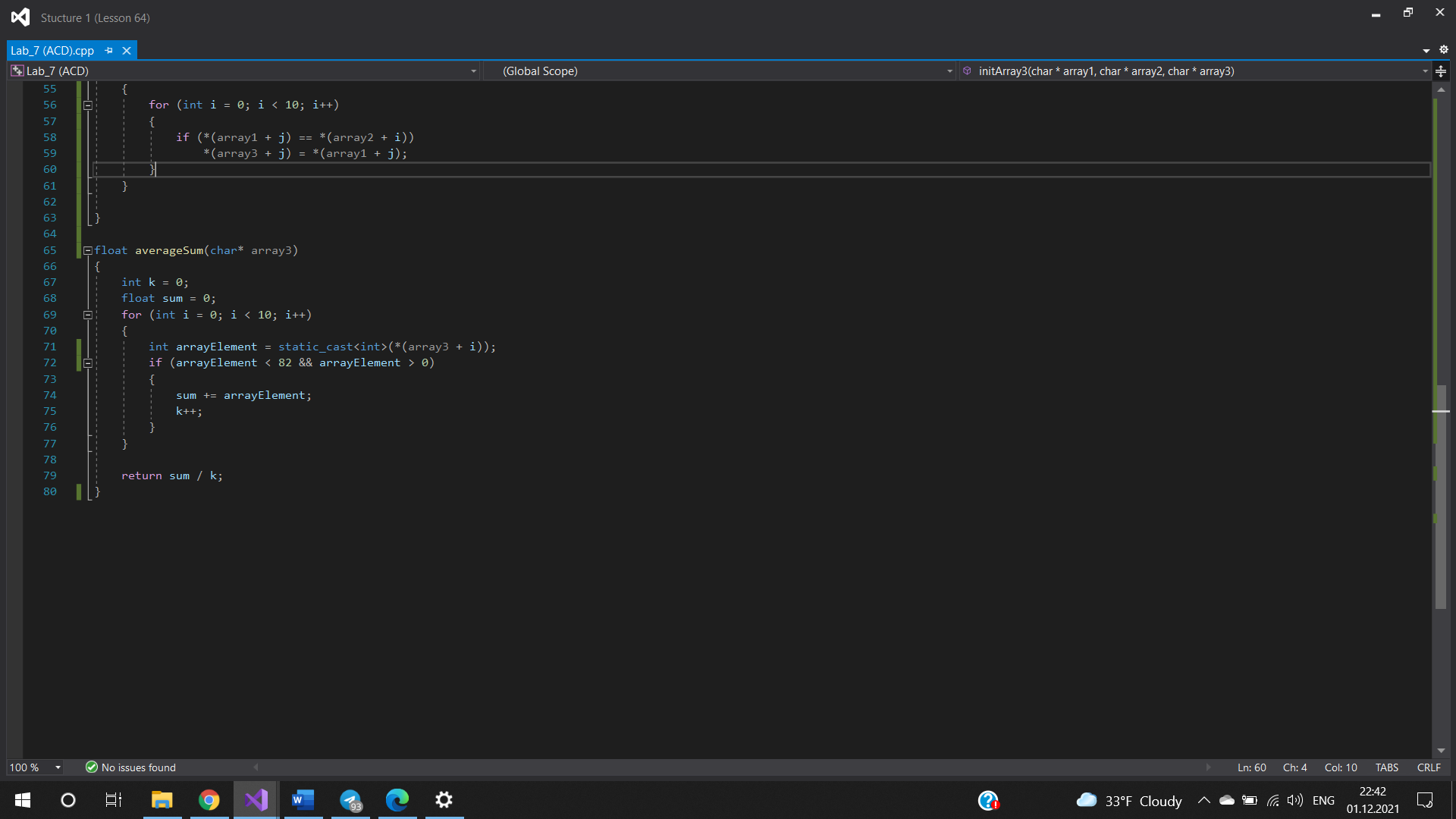


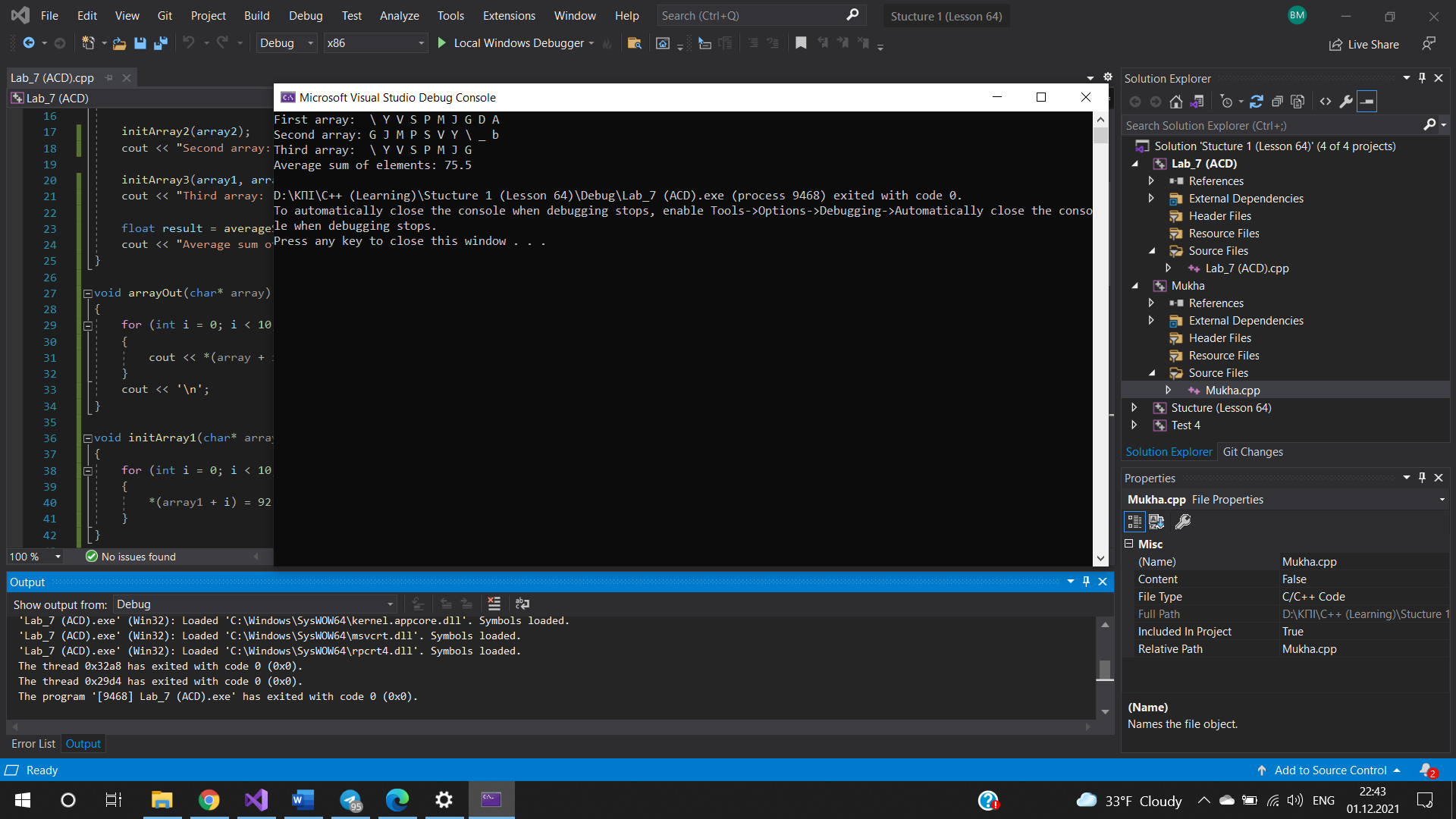


Код програми:









**Випробування алгоритму:**

|  |  |
| --- | --- |
| **Блок** | **Дія** |
|  | Початок |
| 1 | Array1[10] |
| 2 | Array2[10] |
| 3 | Array3[10] |
| 4 | **initArray1**(array1) |
| 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 | **initArray2**(array2) |
| 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 | **initArray2**(array2) |
| 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 | **averageSum**(array3) |
| 30 | **Для і від 1 до 9** |
| 31 | **Якщо** array3[i] < 82 True |
| 32 | Sum = sum + array3[i] |
| 33 | K = k + 1 |
| 38 | **Виведення result** |
|  | **Кінець** |

**Висновки:**

На цій практичній ми дослідили методи послідовного пошуку у впорядкованих і невпорядкованих 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.