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 № 9 з дисципліни

«Алгоритми та структури даних-1.

Основи алгоритмізації»

«Дослідження алгоритмів обходу масивів»   
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 9  
Дослідження алгоритмів обходу масивів**

**Мета –** дослідити алгоритми обходу масивів, набути практичних навичок використання цих алгоритмів під час складання програмних специфікацій.

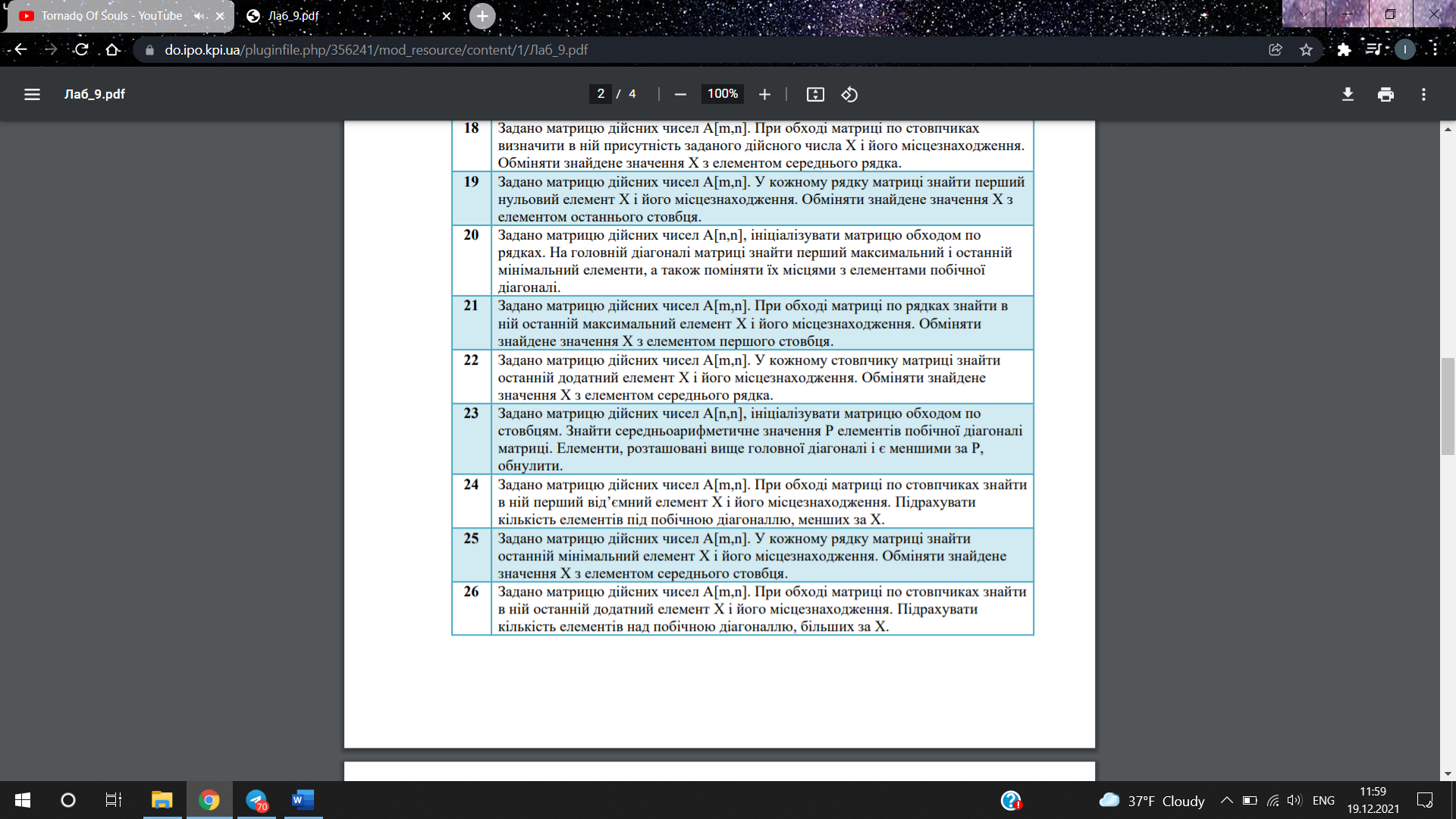
**Варіант 22**

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

1. Опису змінної індексованого типу (двовимірного масиву) згідно з варіантом.

2. Ініціювання змінної, що описана в п.1 даного завдання.

3. Обчислення змінної, що описана в п.1, згідно з варіантом.



**Постановка задачі**

Спочатку ми створюємо двовимірний масив із заданою кількістю рядків і стовпців. Ініціалізація цієї змінної відбувається випадково числами з проміжку [-9, 9] за допомогою підпрограми initMatrix(), при цьому обходимо матрицю «змійкою» по стовпцях. Далі за допомогою підпрограми outMatrix() виводимо матрицю в консоль для наглядності. Після цього за допомогою підпрограми operarion() виводимо в консоль останні додатні елементи і їх положення в матриці. Далі ще раз виводимо всю матрицю щоб побачити зміни в положенні додатних елементів та середніх елементів рядка.

**Побудова математичної моделі**

Складемо таблицю змінних

|  |  |  |  |
| --- | --- | --- | --- |
| **Змінна** | **Тип** | **Ім’я** | **Призначення** |
| Двовимірний масив | Цілий | *matrix* | Проміжне значення |
| Кількість рядків | Цілий | *rows* | Початкове значення |
| Кількість стовпців | Цілий | *columns* | Початкове значення |
| Підпрограма для ініціалізації матриці | void | *initMatrix* | Ініціалізація матриці |
| Підпрограма для виведення матриці | void | *outMatrix* | Виведення матриці в консоль |
| Підпрограма для обробки матриці за варіантом | Дійсний | *operation()* | Обробка матриці за варіантом |

**initMatrix –** підпрограма яка заповнює матрицю за допомогою проходу «змійкою» по стовпцях випадковими числами з проміжку [-9; 9].

**outMatrix** – підпрограма, яка поелементно виводить в консоль дані з матриці за допомогою 2 ітераційних циклів.

**operation() -** підпрограма яка виводить в консоль значення та позицію останнього додатного елемента кожного стовпця та міняє їх з середнім елементом рядків.

**Rand**(a, b) – функція яка генерує випадкове число на проміжку [a, b]

**Розв’язання**

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

*Крок 1*. Визначаємо основні дії  
*Крок 2.* Введення даних  
*Крок 3.* Створення й ініціалізація матриці  
*Крок 4.* Виведення матриці  
*Крок 5.* Обробка матриці

**Псевдокод**

*крок 1***початок**

введення даних  
створення й ініціалізація матриці  
виведення матриці  
обробка матриці

**кінець**

*крок 2*

**початок**

**введення** rows, columns  
створення й ініціалізація матриці  
виведення матриці  
обробка матриці

**кінець**

*крок 3*

**початок**

**введення** rows, columns  
matrix[rows, columns]  
initMatrix(matrix, rows, columns)  
виведення матриці  
обробка матриці

**кінець**

*крок 4*

**початок**

**введення** rows, columns  
matrix[rows, columns]  
initMatrix(matrix, rows, columns)  
outMatrix(matrix, rows, columns  
обробка матриці

**кінець**

*крок 5*

**початок**

**введення** rows, columns  
matrix[rows, columns]  
initMatrix(matrix, rows, columns)  
outMatrix(matrix, rows, columns)  
operation(matrix, rows, columns)  
outMatrix(matrix, rows, columns)

**кінець**

**Псевдокод підпрограм:**

**initMatrix(**matrix, rows, columns**)**

k = 1 **для** j **від** 1 **до** columns **повторити**

**якщо** k > 0

**то для** і **від** 1 **до** rows **повторити** matrix[i, j] = rand(-9, 9)  
 **інакше для** і **від** rows **до** 1 **повторити з кроком** -1  
 matrix[i, j] = rand(-9, 9)  
 k = k \* -1

**все повторити**

**outMatrix(**matrix, rows, columns)

**для** і **від** 1 **до** rows **повторити**

**для** j **від** 1 **до** columns **повторити**

**виведення** matrix[i, j]

**все повторити**

**виведення ‘\n’**

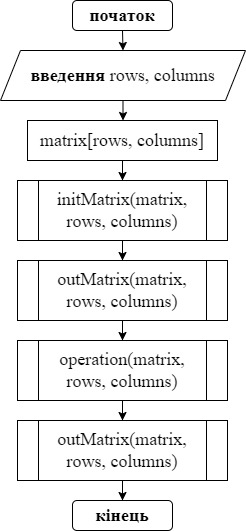
**все повторити**

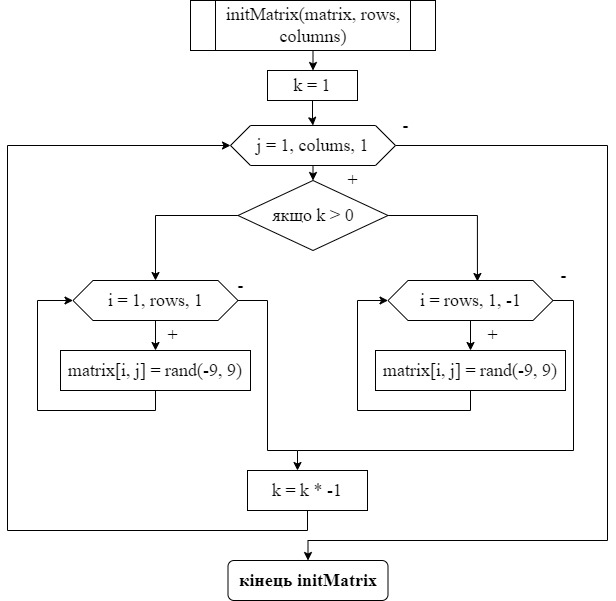
**operation(**matrix, rows, columns**)**

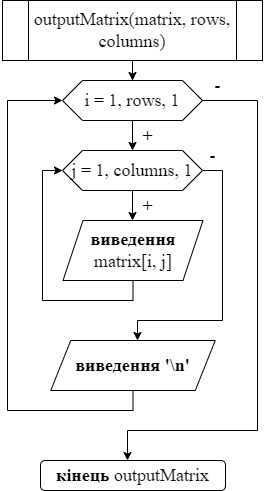
middleRow = (row / 2) + 1  
**для** j **від** 1 **до** columns **повторити**

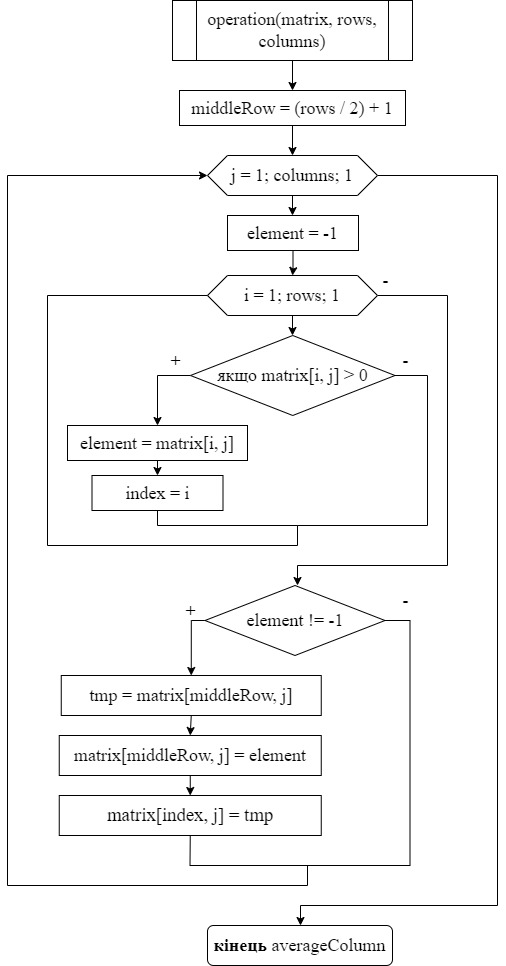
element = -1  
 **для** і **від** 1 **до** rows **повторити  
 якщо** matrix[i, j] > 0  
 element = matrix[i, j]  
 index = i  
 **все якщо  
 все повторити  
 якщо** element != -1  
 tmp = matrix[middleRow, j]  
 matrix[middleRow, j] = element  
 matrix[index, j] = tmp  
 **все якщо**

**все повторити**

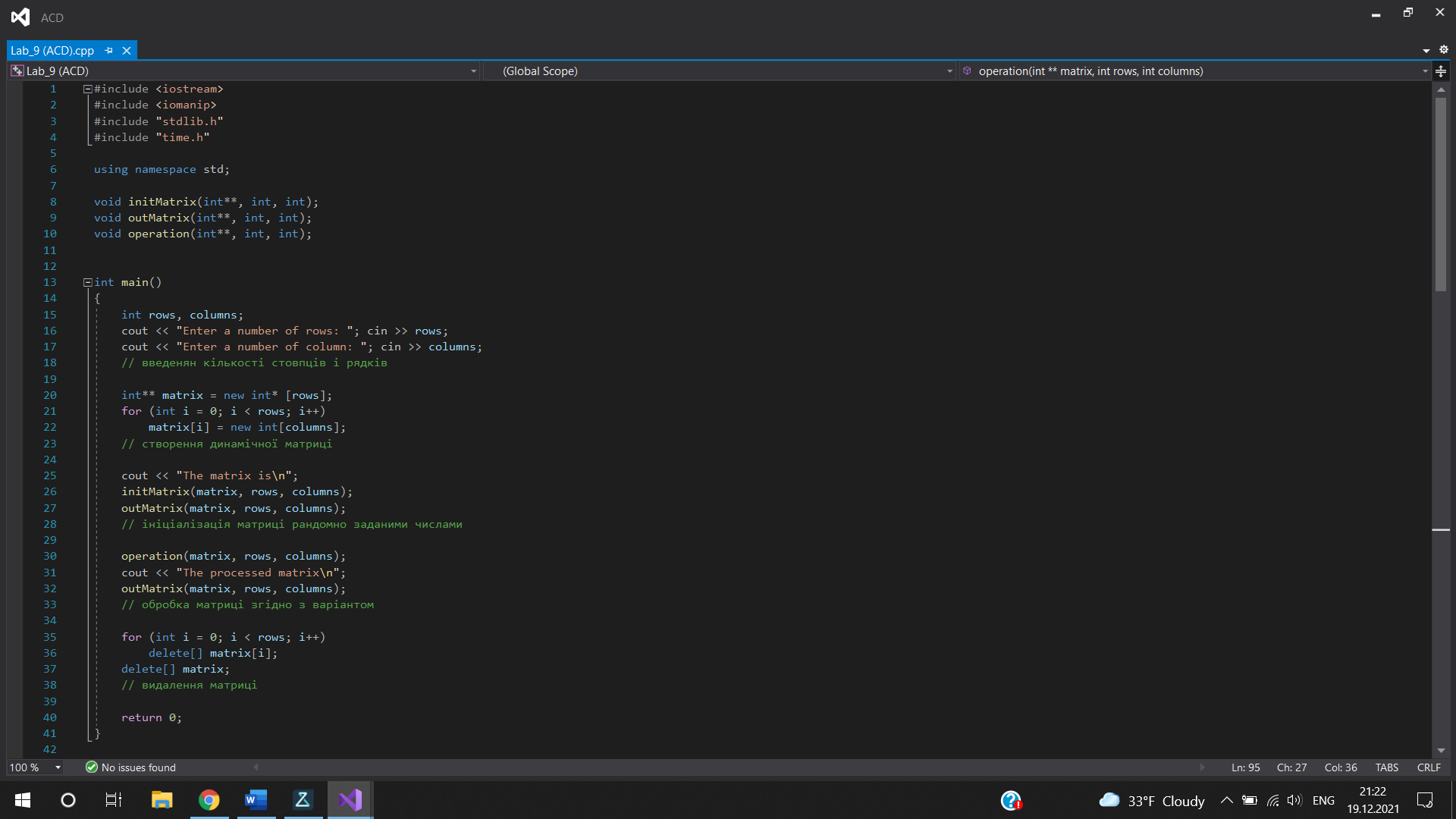
**Блок-схема  
**

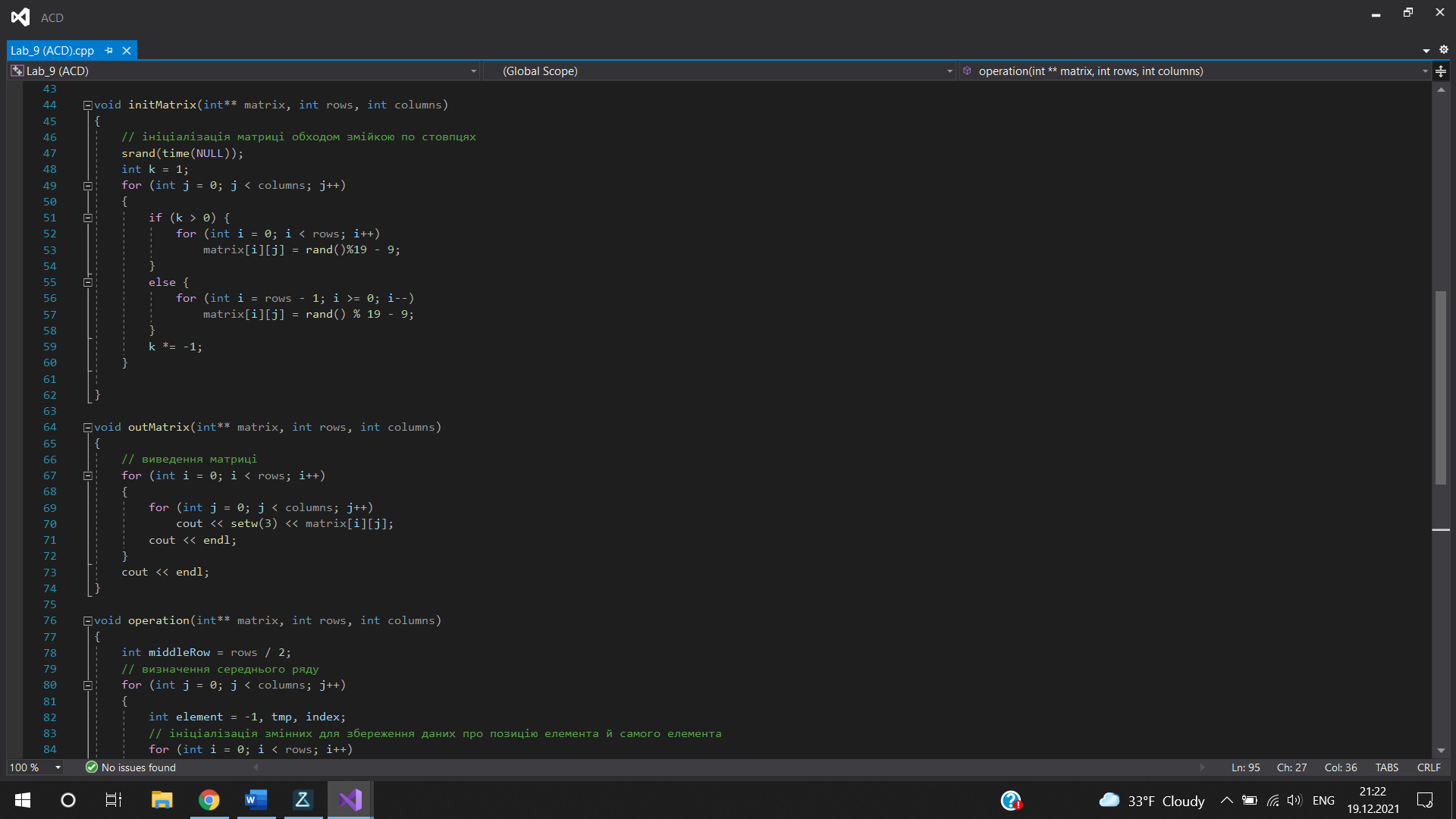


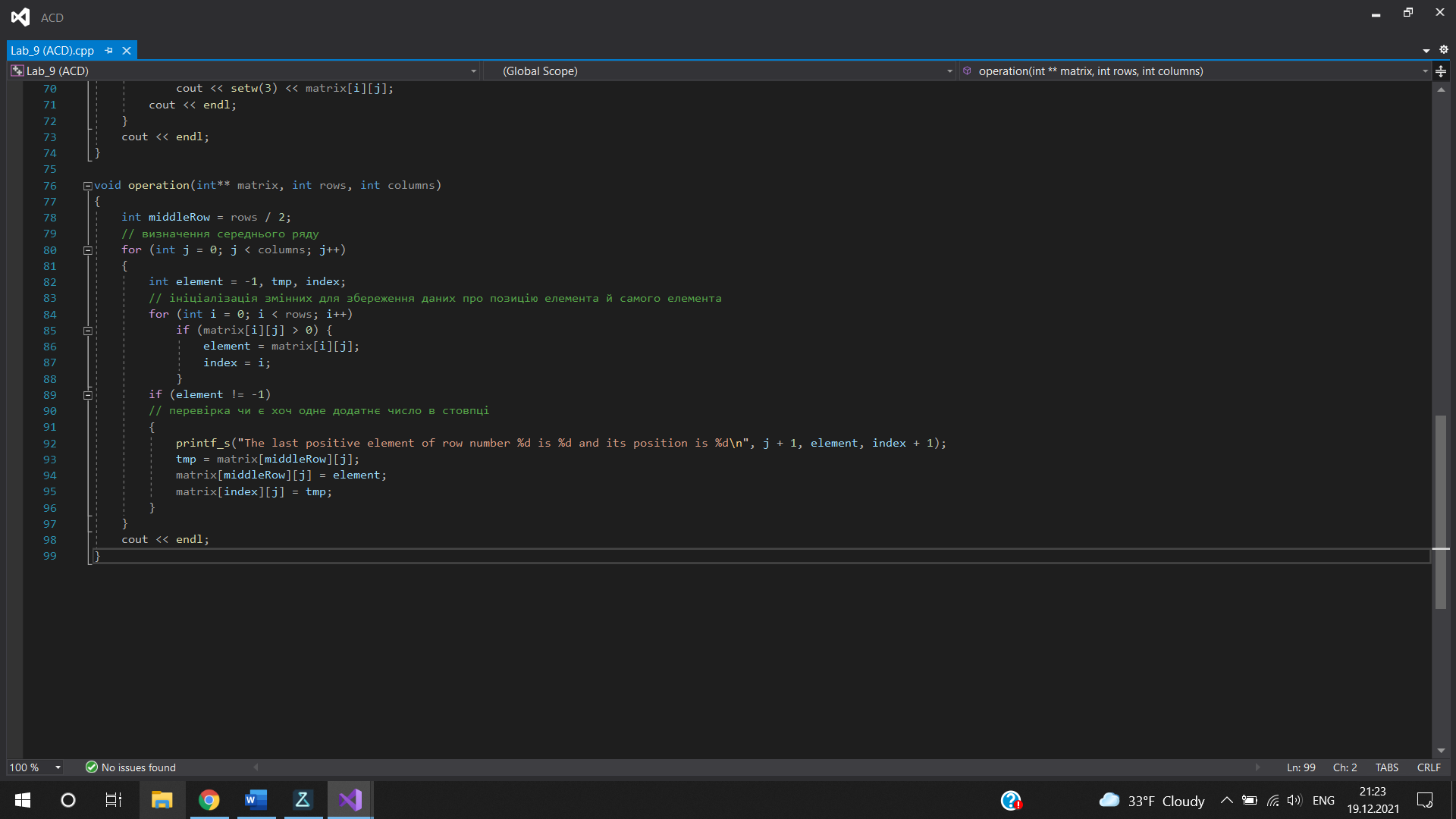




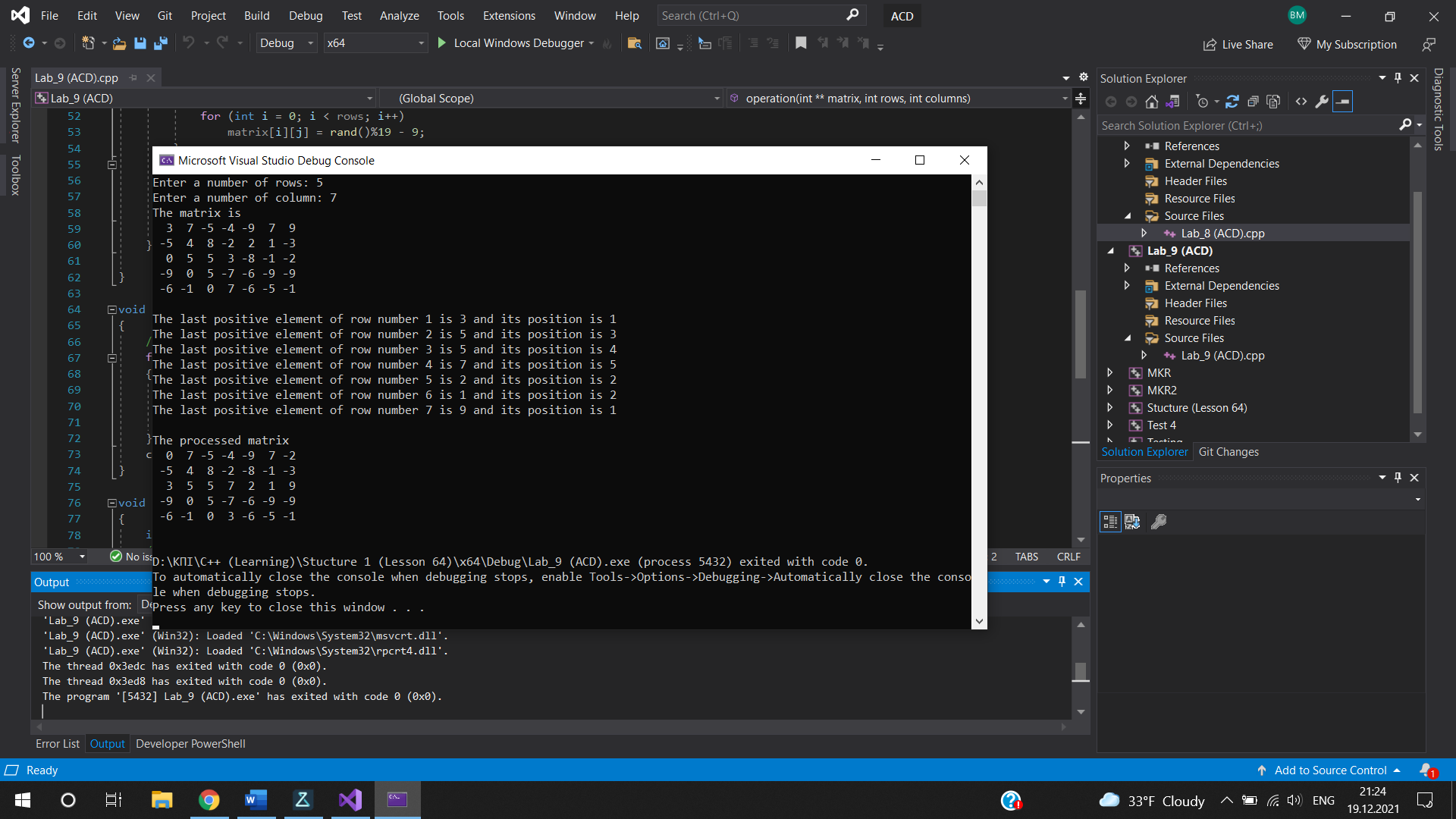
Код програми:

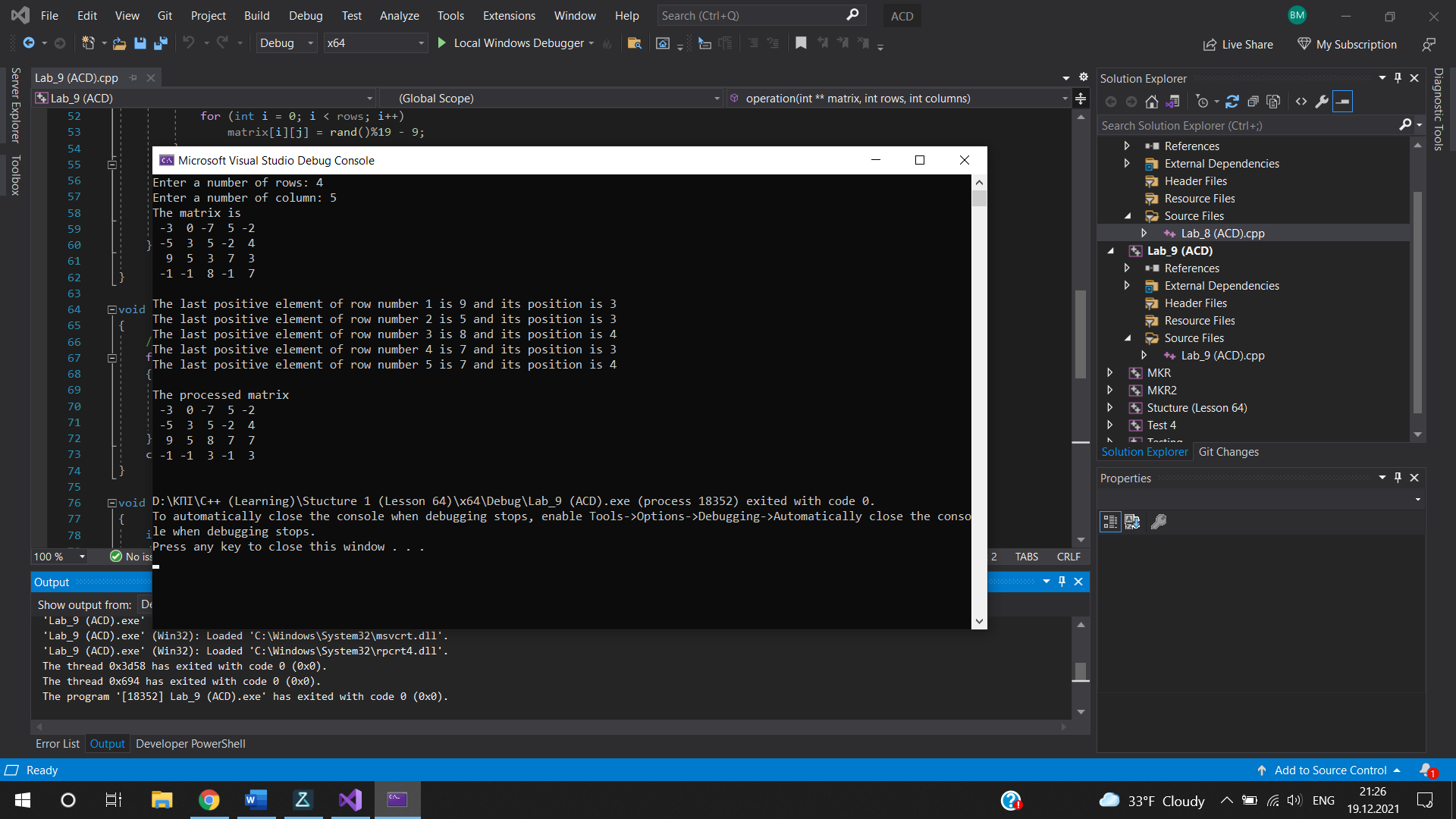






**Випробування алгоритму:**





**Висновки:**

На цій практичній ми дослідити алгоритми обходу матриць, набули практичних навичок використання цих алгоритмів під час складання програмних специфікацій. Також зробили постановку задачі склали матмодель написали псевдокод та намлювали блок-схему. Отримали очікуваний результат.