**Варіант 8**

**Постановка завдання**

Написати функцію транспонування квадратної матриці (тобто повороту початкової матриці на 90°). З її допомогою визначити чи є задана матриця симетричною. (Матриця називається симетричною, якщо транспонована матриця дорівнює початковій).

**Текст програми**

**#include<stdio.h>**

**#include<math.h>**

const int SIZE1 = 100;

const int SIZE2 = 100;

int rows;

int columns;

void transpMatrix(int\*\* arr);

void main(int argc, char\* argv[])

{

printf("Enter number of rows and columns in your matrix:\t");

scanf\_s("%d", &columns);

rows = columns;

int \*\*arr;

arr = new int\*[rows];

for (int i = 0; i < rows; i++)

{

arr[i] = new int[columns];

}

printf("Enter elements of array\n");

for (int i = 0; i < rows; i++)

{

printf("Enter elements of row %d\n", i+1);

for (int j = 0; j < columns; j++)

{

scanf\_s("%d", &arr[i][j]);

}

}

printf("---------------------------\n");

//int arr[SIZE1][SIZE2] = {};

transpMatrix(arr);

system("pause");

}

void transpMatrix(int\*\* arr)

{

int equalValues = 0;

for (int i = 0; i < rows; i++)

{

for (int j = 0; j < columns; j++)

{

if (arr[j][i] == arr[i][j])

{

equalValues++;

}

printf("%d\t", arr[j][i]);

}

printf("\n");

}

if (equalValues == pow(columns, 2))

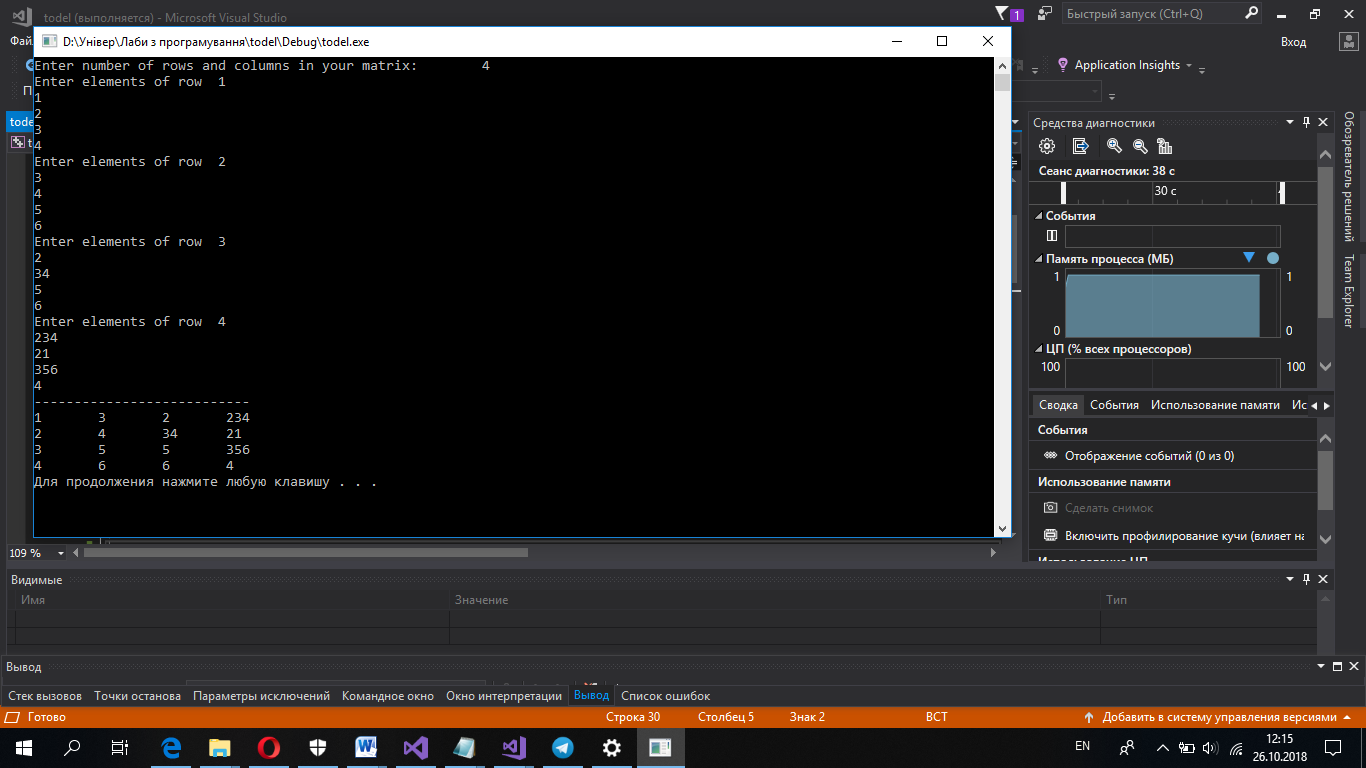
{

printf("Matrix is simetric\n");

}

}

Результат роботи при вказанні несиметричної матриці



Результат роботи при вказанні симетричної матриці

