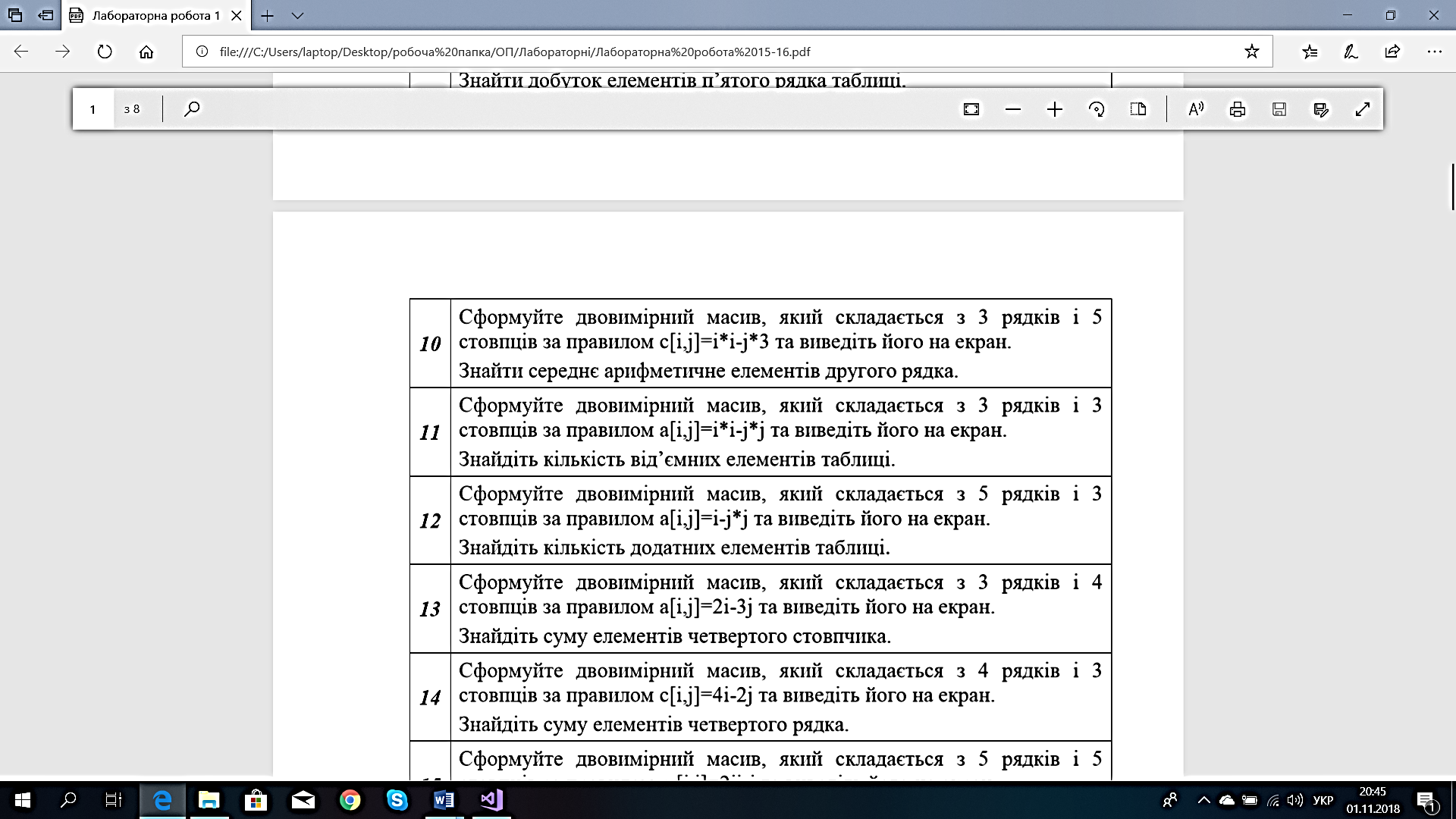
**ЛАБОРАТОРНА РОБОТА № 15-16**

**Опрацювання двовимірних масивів.**

***Мета:*** набуття практичних навичок роботи з двовимірними масивами.

**Хід роботи:**

**Завдання 1**: Написати програму згідно варіанту. Результати вивести на екран у зручному для сприйняття вигляді.



Умова до завдання № 1

Лістинг програми:

#include "pch.h"

#include <iostream>

#include <locale.h>

#include <math.h>

#define I 3

#define J 5

int main()

{

setlocale(LC\_ALL, "Rus");

int arr[I][J], i, j, average = 0;

for (i = 0; i < I; i++) {

for (j = 0; j < J; j++) {

arr[i][j] = i \* i - j \* 3;

printf("%4d", arr[i][j]);

if (i == 1) {

average += arr[i][j];

}

}

printf("\n");

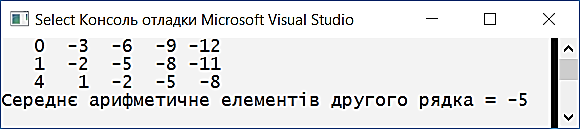
}

average /= 5;

printf("Cереднє арифметичне елементiв другого рядка = %d\n", average);

}

Результат виконання програми:



Результат виконання програми до завдання №1

**Завдання 2:** Сформуйте двовимірний масив b дійсних чисел з n=5 рядками і m=5 стовпцями. Виведіть масив на екран у вигляді таблиці, задайте формати виведення чисел з двома числами після крапки. Виконайте завдання:

1) визначте максимальний елемент і його індекси;

2) визначте мінімальний елемент і його індекси;

3) обчисліть середньоарифметичні значення елементів кожного рядка;

4) обчисліть добуток елементів головної діагоналі;

5) обчисліть суму елементів головної діагоналі;

6) обчисліть суму елементів під головною діагоналлю (підказка: i<j).

Лістинг програми:

#include "pch.h"

#include <iostream>

#include <time.h>

#include <stdlib.h>

#define I 5

#define J 5

int main()

{

srand(time(0));

int iMax[2], iMin[2];

float arr[I][J], max = -1e+7, min = 1e+7, average[J], temp, sum = 0, mult = 1, sum2 = 0;

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

{

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

arr[i][j] = -5 + rand() % 10, printf(" %5.2f", arr[i][j]);

printf("\n");

}

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

{

temp = 0;

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

{

arr[i][j] > max ? max = arr[i][j], iMax[0] = i, iMax[1] = j : 0;

arr[i][j] < min ? min = arr[i][j], iMin[0] = i, iMin[1] = j : 0;

temp += arr[i][j];

}

average[i] = temp / I / 1.0;

}

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

{

sum += arr[i][i];

mult \*= arr[i][i];

}

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

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

j < i ? sum2 += arr[i][j] : 0;

printf("\n\n");

printf("\nMax: %.2f, id: %d %d", max, iMax[0], iMax[1]);

printf("\nMin: %.2f, id: %d %d", min, iMin[0], iMin[1]);

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

printf("\nAverage %d: %.2f", i + 1, average[i]);

printf("\nMain diagonal sum: %.2f", sum);

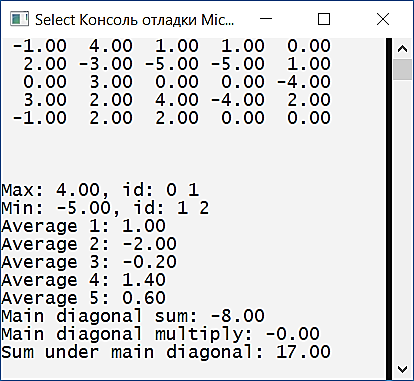
printf("\nMain diagonal multiply: %.2f", mult);

printf("\nSum under main diagonal: %.2f", sum2);

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

}

Результат виконання програми:



Результат виконання програми до завдання №2

**Завдання 3:** Розробити програму, дотримуючись таких вимог:

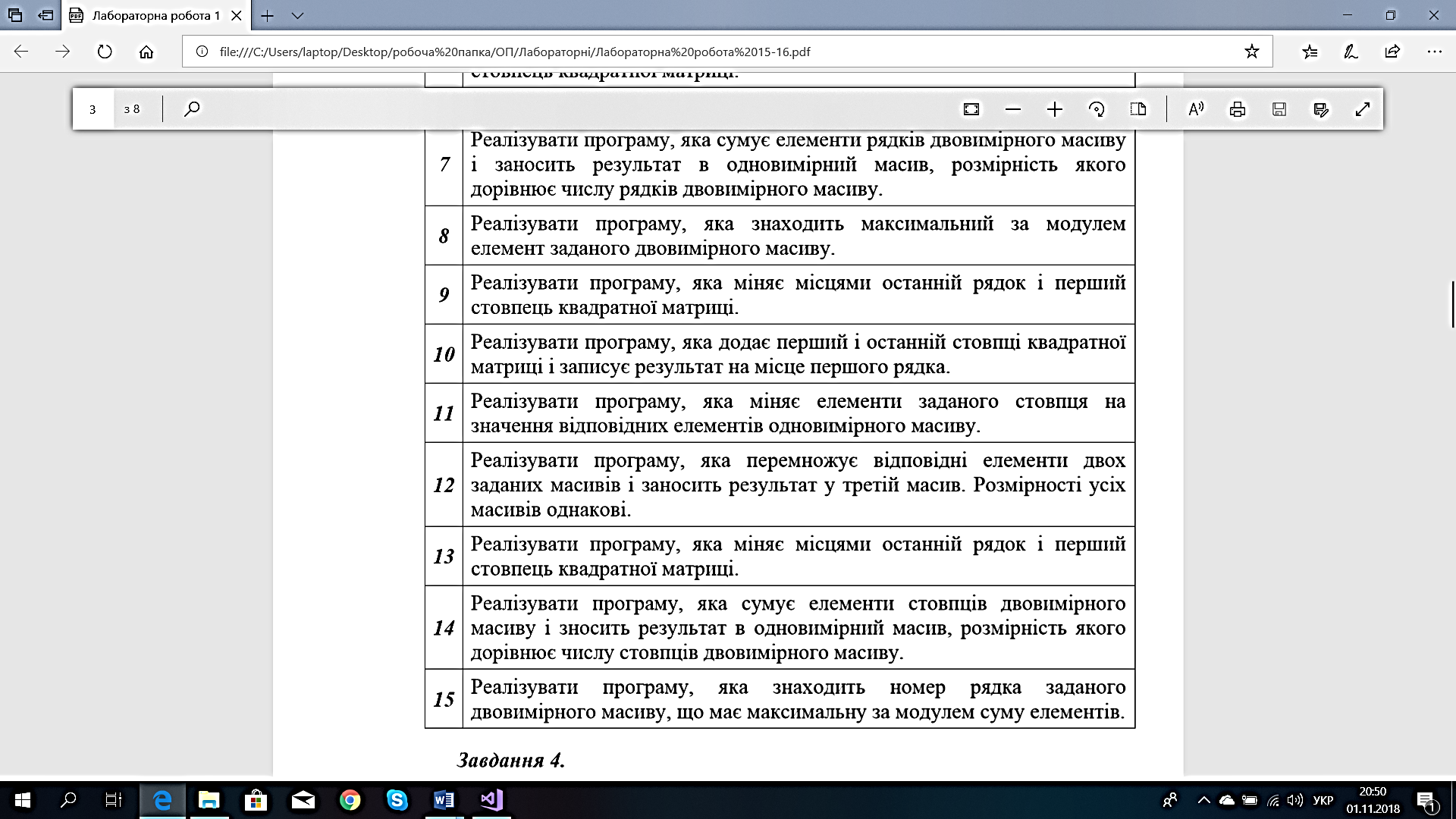
• використовувати статичні масиви;

• максимальні розміри масиву [N][M] – статичні константи;

• реальні розміри масиву n i m (n<N, m<M) – ввести з клавіатури (при цьому здійснювати перевірку правильності введення даних);

• елементи масиву – псевдовипадкові числа, що згенеровані на інтервалі [a, b], де a і b(a<b) вводяться з клавіатури;

• усі вхідні дані і елементи масиву виводити на екран.



Умова до завдання №3

Лістинг програми:

#include "pch.h"

#include <iostream>

#include <time.h>

#include <stdlib.h>

#include <locale.h>

#define M 20

#define N 20

int main()

{

setlocale(LC\_ALL, "Rus");

srand(time(0));

int i, j, m, n, arr[N][M], is = 0, a, b;

do {

printf("Введите n:"), scanf\_s("%d", &n);

printf("Введите m:"), scanf\_s("%d", &m);

printf("Введите a:"), scanf\_s("%d", &a);

printf("Введите b:"), scanf\_s("%d", &b);

a > b || n > N || m > M || n < 1 || m < 1 ? printf(" Error\n"), is = 1 : 0;

} while (is);

printf("\n\n");

for (i = 0; i < n; i++)

{

for (j = 0; j < m; j++) {

arr[i][j] = a + rand() % (b - a + 1), printf("%3d", arr[i][j]);

}

printf("\n");

}

printf("\n");

for (i = 0; i < n; i++) {

arr[0][i] = arr[i][0] + arr[i][n - 1];

}

for (i = 0; i < n; i++)

{

for (j = 0; j < m; j++) {

printf("%3d", arr[i][j]);

}

printf("\n");

}

printf("\n\n");

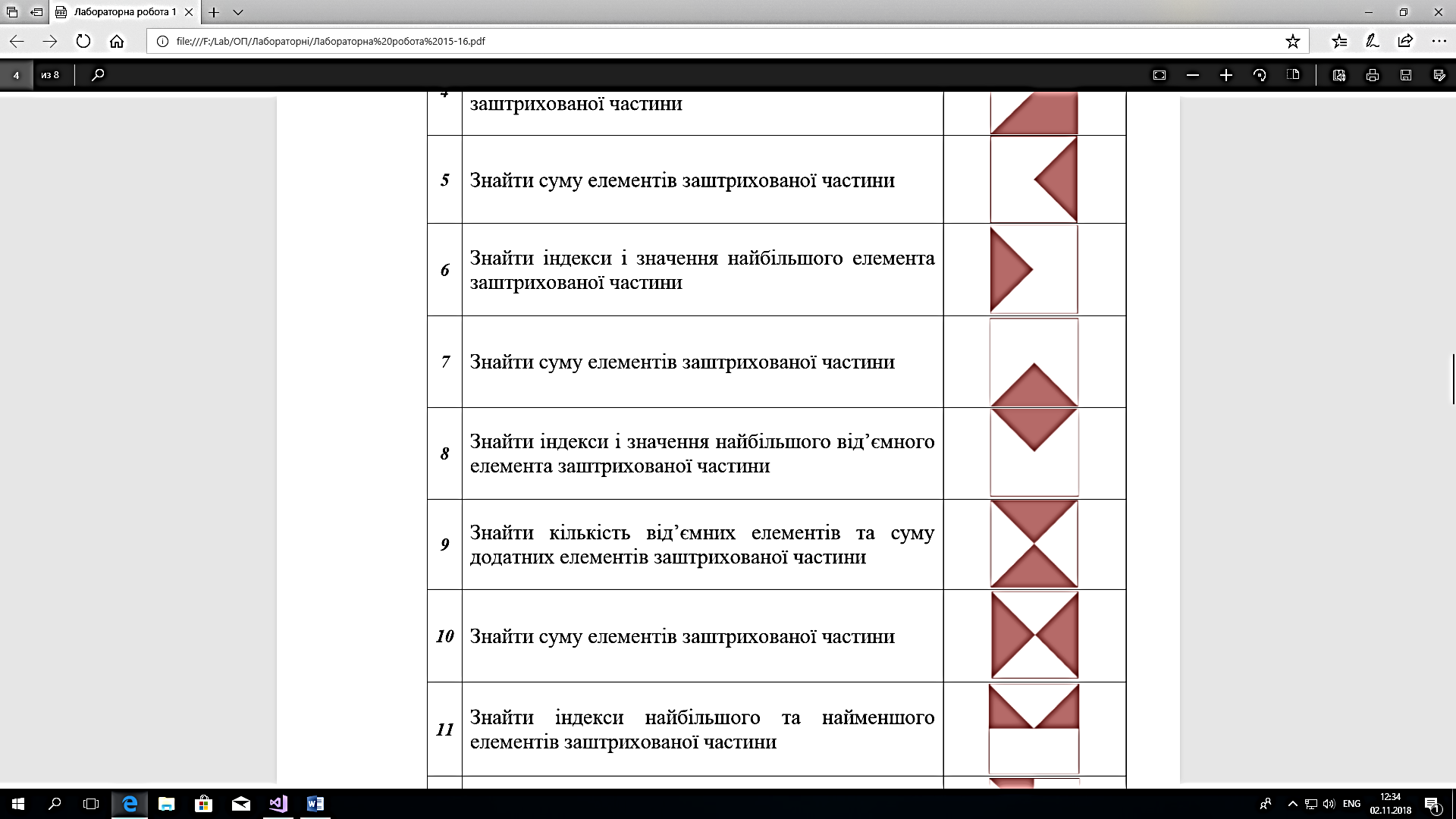
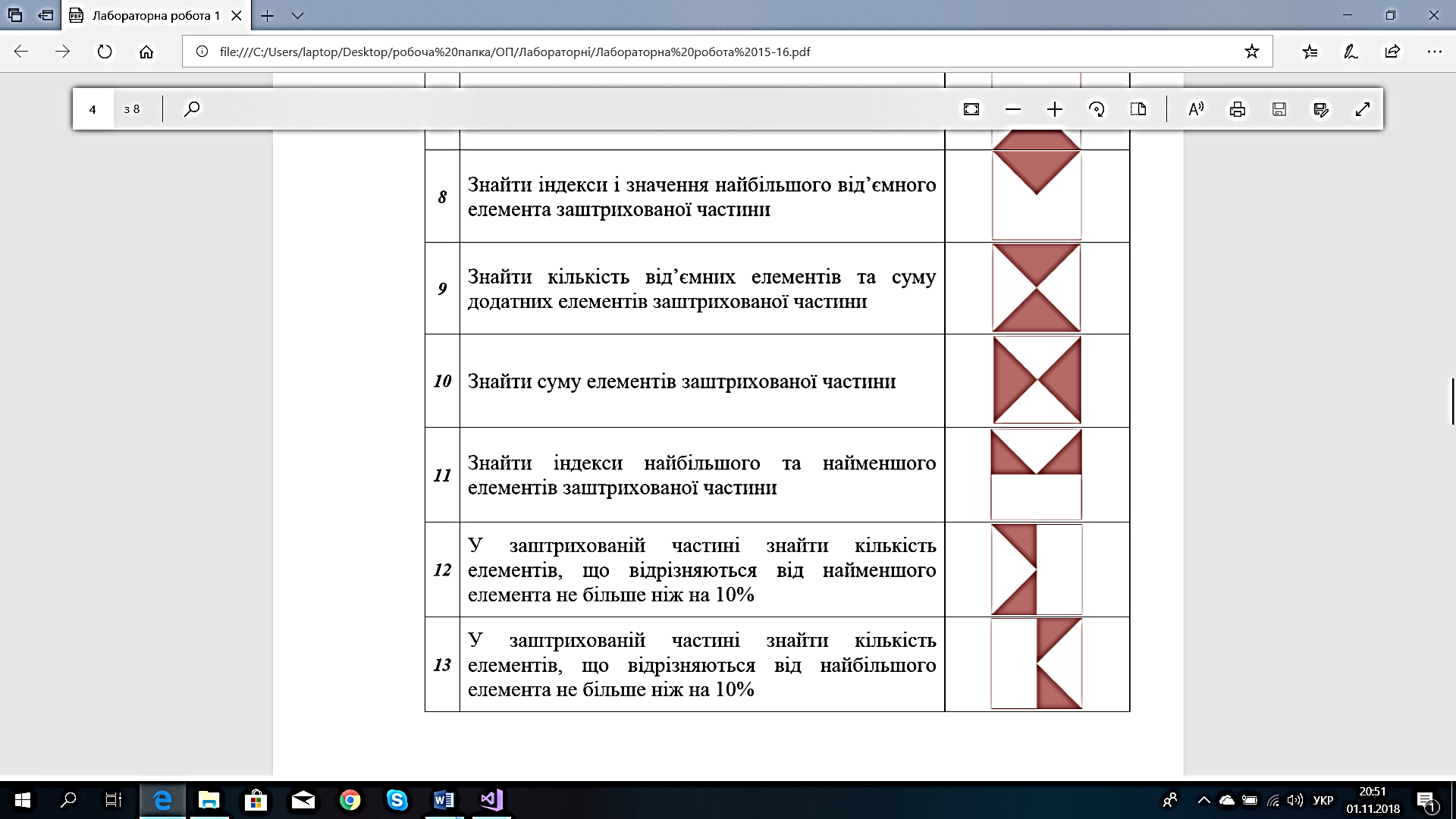
}

Результат виконання програми:



Результат виконання програми до завдання №3

**Завдання 4:**



Умова до завдання №4

Лістинг програми:

#include "pch.h"

#include <iostream>

#include <time.h>

#include <stdlib.h>

#include <locale.h>

#define M 5

#define N 5

int main()

{

setlocale(LC\_ALL, "Rus");

srand(time(0));

int arr[M][N], sum = 0;

for (int i = 0; i < M; i++) {

for (int j = 0; j < N; j++) {

arr[i][j] = rand() % 10;

printf("%3d", arr[i][j]);

}

printf("\n");

}

for (int j = 0; (M % 2) == 0 ? j < M / 2 : j < (M / 2) + 1; j++) {

for (int i = j; i < N - j; i++) {

sum += arr[i][j];

}

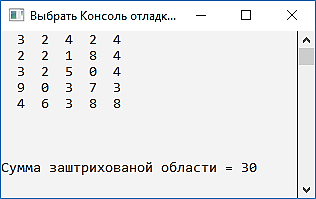
printf("\n");

}

printf("Сумма заштрихованой области = %d\n", sum);

}

Результат виконання програми:



Результат виконання програми до завдання №4

***Завдання на самостійну роботу:***

**Задача про вибори.** Нехай шість населених пунктів позначені номерами від 1 до 6 (змінна k), а п’ять кандидатів – номерами від 1 до 5 (змінна n). Кількість голосів, набраних кандидатами в кожному пункті, визначається за формулою A[k] = rand()%10i + 50, де i – номер варіанта. Вивести на екран таблицю результатів голосування, де значення в рядках – дані з населених пунктів, а в стовпцях – дані по конкретних кандидатах.

***Знайти :***

1. Підсумкові результати кожного кандидата. (створити одномірний масив із сум значень стовпців таблиці).

2. Хто з кандидатів набрав максимальну, а хто мінімальну кількість голосів у 4 населеному пункті?

3. Яку кількість голосів було подано за 1 і 3 кандидатів у всіх населених пунктах?

4. У яких населених пунктах 2-й та 4-тий кандидати набрали максимальну кількість голосів?

5. Скільки виборців взяли участь у голосуванні в кожному населеному пункті?

6. Хто з кандидатів має максимальний рейтинг?

7. У яких населених пунктах 1 кандидат набрав максимальну кількість голосів?

9. У якому населеному пункті проголосувало найбільше виборців?

10. Хто з кандидатів набрав найбільшу кількість голосів у 2 і 3 населених пунктах?

11. У якому населеному пункті 1 кандидат набрав мінімальну кількість голосів, а в якому максимальну?

12. У кого серед другого, четвертого і п’ятого кандидатів найвищий рейтинг?

13. Хто набрав максимальну, а хто мінімальну кількість голосів у 1 населеному пункті?

14. У яких населених пунктах 1 і 5 кандидати набрали більше 100 голосів?

15. У яких населених пунктах кількість виборців не перевищила 450?

16. Які кандидати набрали мінімальну кількість голосів у кожному з пунктів?

17. Які кандидати набрали максимальну і мінімальну кількість голосів у 2 і 5 населених пунктах? Лістинг програми:

Лістинг програми:

#include "pch.h"

#include <iostream>

#include <time.h>

#include <stdlib.h>

#define N 6

#define M 5

int main()

{

srand(time(0));

int arr[N][M], sumM[M] = {0};

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

{

printf("%d) | ", i + 1);

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

arr[i][j] = rand() % 130, printf(" %3d |", arr[i][j]);

printf("\n");

}

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

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

sumM[i] += arr[j][i];

printf("\n\n1) \n");

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

printf(" %d", sumM[i]);

int max = -1e+7, iMax, min = 1e+7, iMin;

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

{

max < arr[3][i] ? max = arr[3][i], iMax = i : 0;

min > arr[3][i] ? min = arr[3][i], iMin = i : 0;

}

printf("\n\n2) \n Max: %d.", iMax + 1);

printf("\n Min: %d.", iMin + 1);

printf("\n\n3) \n 1. %d\n 3. %d", sumM[0], sumM[2]);

int max2 = -1e+7, max4 = -1e+7, iMax2, iMax4;

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

{

arr[i][1] > max2 ? max2 = arr[i][1], iMax2 = i : 0;

arr[i][3] > max4 ? max4 = arr[i][3], iMax4 = i : 0;

}

printf("\n\n4) \n 2. in %d-th city\n 4. in %d-th city", iMax2 + 1, iMax4 + 1);

int sumN[N] = {0};

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

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

sumN[j] += arr[j][i];

printf("\n\n5) \n");

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

printf(" %d", sumN[i]);

int maxR = 0, iMaxR;

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

maxR < sumM[i] ? maxR = sumM[i], iMaxR = i : 0;

printf("\n\n6) \n");

printf(" Max rating has %d. candidate", iMaxR + 1);

int rate[N] = {0};

for (int i = 0, k = 0, is; i < N; i++, k++)

{

maxR = arr[i][0];

is = 1;

for (int j = 1; j < M; j++)

maxR > arr[i][j] ? 0 : is = 0;

if (is)

rate[k] = 1;

}

printf("\n\n7) \n In:");

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

if (rate[i])

printf(" %d)", i+1);

max = -1e+7;

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

max < sumN[i] ? max = sumN[i], iMax = i:0;

printf("\n\n9) \n In: %d) city", iMax + 1);

max2 = -1e+7, max4 = -1e+7;

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

{

max2 < arr[1][M] ? max2 = arr[1][M], iMax2 = i : 0;

max4 < arr[2][M] ? max4 = arr[2][M], iMax4 = i : 0;

}

printf("\n\n10) \n In 2) city %d.\n In 3) city %d.", iMax2 + 1, iMax4 + 1);

max = -1e+7, min = 1e+7;

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

{

max < arr[i][0] ? max = arr[i][0], iMax = i : 0;

min > arr[i][0] ? min = arr[i][0], iMin = i : 0;

}

printf("\n\n11) \n Min in %d) city\n Max in %d) city", iMin + 1, iMax + 1);

sumM[1] > sumM[3] ? iMax = 2 : sumM[3] > sumM[4] ? iMax = 4 : iMax = 5;

printf("\n\n12) \n Max rating(of 2, 4, 5) has %d. candidate", iMax);

max = -1e+7, min = 1e+7;

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

{

max < arr[0][i] ? max = arr[0][i], iMax = i : 0;

min > arr[0][i] ? min = arr[0][i], iMin = i : 0;

}

printf("\n\n13) \n Max rating has %d. candidate\n Min rating has %d. candidate", iMax+1, iMin+1);

int rate1[M] = { 0 }, rate5[M] = {0};

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

{

arr[i][0] > 100 ? rate1[i] = 1 : 0;

arr[i][4] > 100 ? rate5[i] = 1 : 0;

}

printf("\n\n14) \n 1. > 100 in:");

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

if (rate1[i])

printf(" %d)", i+1);

printf(" cities \n 5. > 100 in:");

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

if (rate5[i])

printf(" %d)", i+1);

printf(" cities");

printf("\n\n15) \n In:");

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

if (sumN[i] <= 450)

printf(" %d)", i + 1);

printf(" cities < 450");

int is = 1, iMin1[M];

max = -1e+7, min = 1e+7;

for (int i = 0, k = 0; i < N; i++)

{

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

arr[i][j] < min ? min = arr[i][j], iMin1[k] = j:0;

k++;

}

for (int i = 0; i < N - 1; i++)

iMin1[i] == iMin1[i + 1] ? 0 : is = 0;

printf("\n\n16) \n ");

if (is)

printf(" %d.", iMin1[0] + 1);

else

printf(" No one");

printf(" - has minimal rating in all cities");

max = -1e+7, min = 1e+7;

int max1 = -1e+7, min1 = 1e+7, iMax3, iMin3;

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

{

arr[1][i] > max ? max = arr[1][i], iMax = i : 0;

arr[1][i] < min ? min = arr[1][i], iMin = i : 0;

arr[4][i] > max1 ? max1 = arr[4][i], iMax3 = i : 0;

arr[4][i] < min1 ? min1 = arr[4][i], iMin3 = i : 0;

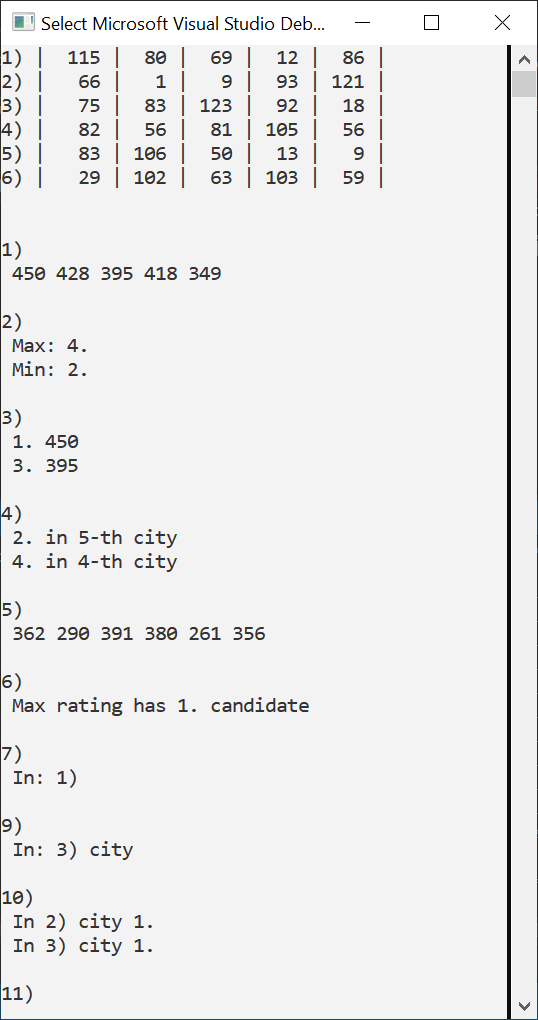
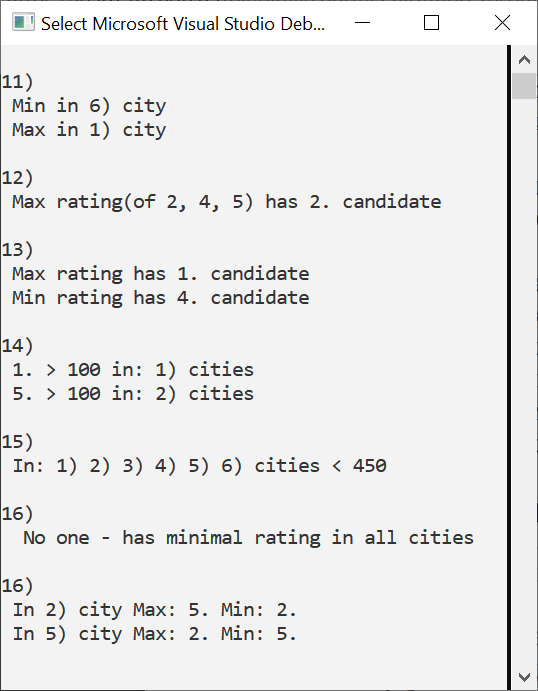
}

printf("\n\n16) \n In 2) city Max: %d. Min: %d.\n", iMax +1, iMin+1);

printf(" In 5) city Max: %d. Min: %d.\n\n\n", iMax3 + 1, iMin3 + 1);

}

Результат виконання програми:

Результат виконання програми до завдання на самотійну роботу

***Висновок:*** в ході виконання лабораторної роботи набуто практичних навичок роботи з двовимірними масивами.