**мІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**нАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «лЬВІВСЬКА ПОЛІТЕХНІКА»**

**ІНСТИТУТ КОМП’ЮТЕРНИХ НАУК ТА ІНОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

**Кафедра ІСМ**

|  |
| --- |
| **2020** |



**ЗВІТ**

**Про виконання лабораторної роботи № 7.1**

**« Пошук заданих елементів та**

**впорядкування рядків / стовпчиків матриці »**

**З дисципліни**

**«Алгоритмізація та програмування»**

**Студента групи ІТ-12**

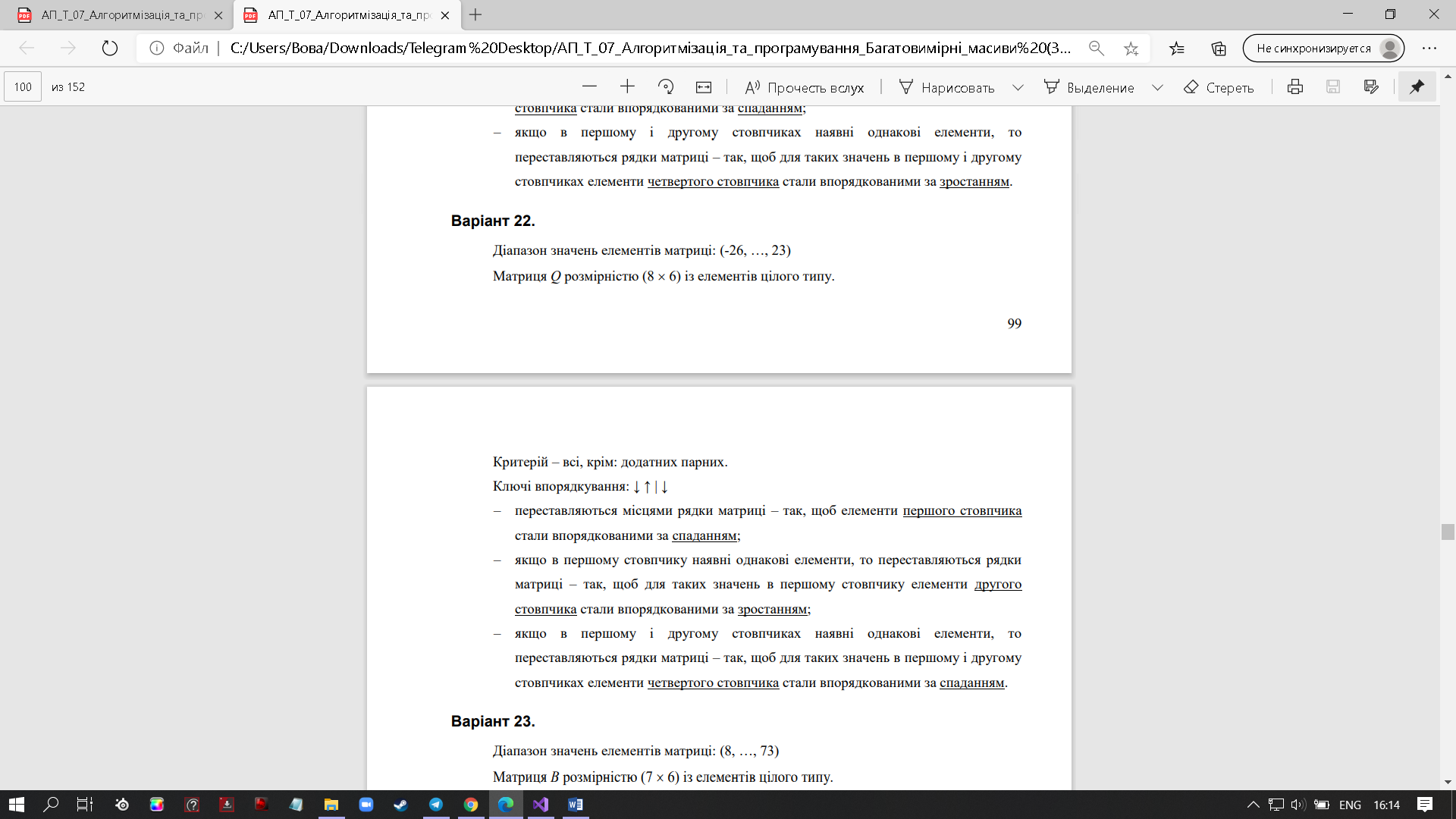
**Шкринди Володимира**

**Мета роботи**

Навчитися обчислювати кількість, суму заданих елементів матриці; а також

впорядковувати матрицю, переставляючи місцями її рядки / стовпчики.

**Умова завдання**



**Текст програми**

// Lab\_7\_1.cpp

// < Шкринда В.В >

// Лабораторна робота № 7.1.

// Пошук заданих елементів та впорядкування рядків / стовпчиків матриці.

// Варіант 22

#include <iostream>

#include <iomanip>

#include <time.h>

using namespace std;

void Create(int\*\* Q, const int rowCount, const int colCount, const int Low,

const int High);

void Print(int\*\* Q, const int rowCount, const int colCount);

void Sort(int\*\* Q, const int rowCount, const int colCount);

void Change(int\*\* Q, const int row1, const int row2, const int colCount);

void Calc(int\*\* Q, const int rowCount, const int colCount, int& S, int& k);

int main()

{

srand((unsigned)time(NULL));

int Low = -26;

int High = 23;

int rowCount = 8;

int colCount = 6;

int\*\* Q = new int\* [rowCount];

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

Q[i] = new int[colCount];

Create(Q, rowCount, colCount, Low, High);

Print(Q, rowCount, colCount);

Sort(Q, rowCount, colCount);

Print(Q, rowCount, colCount);

int S = 0;

int k = 0;

Calc(Q, rowCount, colCount, S, k);

cout << "S = " << S << endl;

cout << "k = " << k << endl;

Print(Q, rowCount, colCount);

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

delete[] Q[i];

delete[] Q;

return 0;

}

void Create(int\*\* Q, const int rowCount, const int colCount, const int Low,

const int High)

{

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

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

Q[i][j] = Low + rand() % (High - Low + 1);

}

void Print(int\*\* Q, const int rowCount, const int colCount)

{

cout << endl;

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

{

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

cout << setw(4) << Q[i][j];

cout << endl;

}

cout << endl;

}

void Sort(int\*\* Q, const int rowCount, const int colCount)

{

for (int i0 = 0; i0 < rowCount - 1; i0++)

for (int i1 = 0; i1 < rowCount - i0 - 1; i1++)

if ((Q[i1][0] < Q[i1 + 1][0])

||

(Q[i1][0] == Q[i1 + 1][0] &&

Q[i1][1] > Q[i1 + 1][1])

||

(Q[i1][0] == Q[i1 + 1][0] &&

Q[i1][1] == Q[i1 + 1][1] &&

Q[i1][3] < Q[i1 + 1][3]))

Change(Q, i1, i1 + 1, colCount);

}

void Change(int\*\* Q, const int row1, const int row2, const int colCount)

{

int tmp;

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

{

tmp = Q[row1][j];

Q[row1][j] = Q[row2][j];

Q[row2][j] = tmp;

}

}

void Calc(int\*\* Q, const int rowCount, const int colCount, int& S, int& k)

{

S = 0;

k = 0;

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

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

if (!(Q[i][j] % 2 == 0 && Q[i][j]> 0))

{

S += Q[i][j];

k++;

Q[i][j] = 0;

}

}

**Unit-test: немає бо лише використовувався void**

**Cруктурна схема:**

**Посилання на git-репозиторій з проектом:**

**Висновок:** на цій лабораторній роботі я навчився