ЗВІТ

про виконання лабораторної роботи № 7.1.1

«Пошук заданих елементів та впорядкування рядків / стовпчиків матриці»

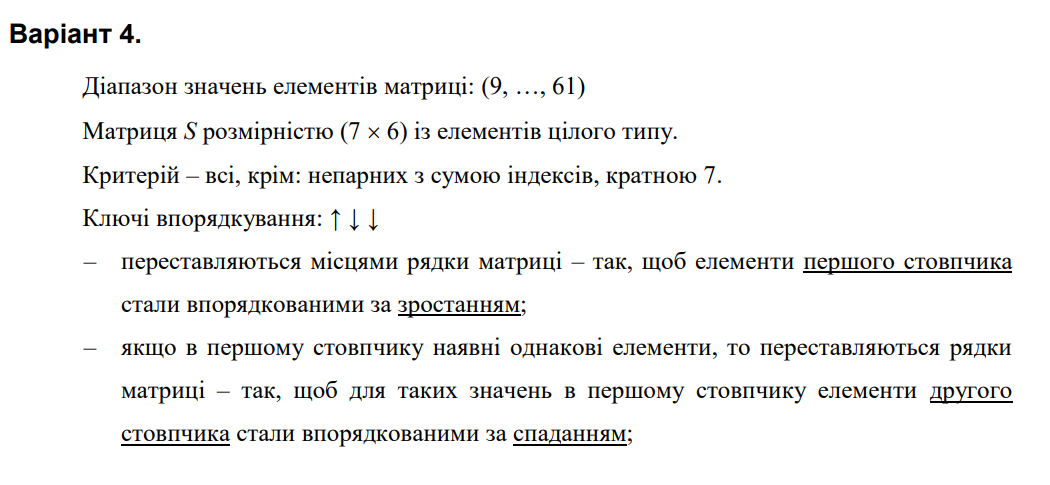
з дисципліни

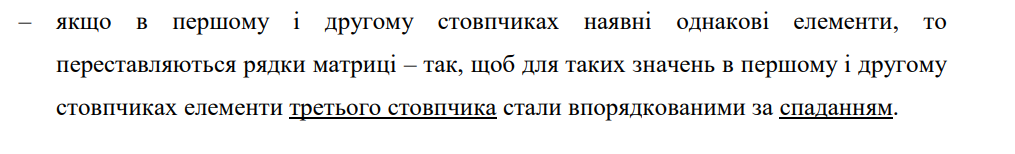
«Алгоритмізація та програмування»

студента групи РІ-11

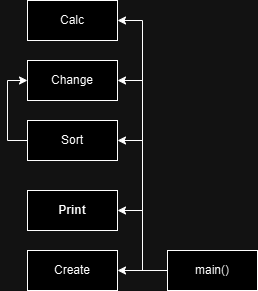
Ганина Дмитра Юрійовича

Умова завдання:





Структурна схема програми:



Текст програми:

// Lab\_7\_1.cpp

// Ганин Дмитро

// Лабораторна робота № 7.1.

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

// Варіант 4

#include <iostream>

#include <iomanip>

#include <time.h>

using namespace std;

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

const int High);

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

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

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

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

int main()

{

srand((unsigned)time(NULL));

int Low = 9;

int High = 61;

int rowCount = 7;

int colCount = 6;

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

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

a[i] = new int[colCount];

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

Print(a, rowCount, colCount);

Sort(a, rowCount, colCount);

Print(a, rowCount, colCount);

int S = 0;

int k = 0;

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

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

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

Print(a, rowCount, colCount);

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

delete[] a[i];

delete[] a;

return 0;

}

void Create(int\*\* a, 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++)

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

}

void Print(int\*\* a, 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) << a[i][j];

cout << endl;

}

cout << endl;

}

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

{

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

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

if ((a[i1][0] > a[i1 + 1][0])

||

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

a[i1][1] < a[i1 + 1][1])

||

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

a[i1][1] == a[i1 + 1][1] &&

a[i1][2] < a[i1 + 1][2]))

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

}

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

{

int tmp;

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

{

tmp = a[row1][j];

a[row1][j] = a[row2][j];

a[row2][j] = tmp;

}

}

void Calc(int\*\* a, 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 ((a[i][j] % 2 != 0) && !(i + j == 7))

{

S += a[i][j];

k++;

a[i][j] = 0;

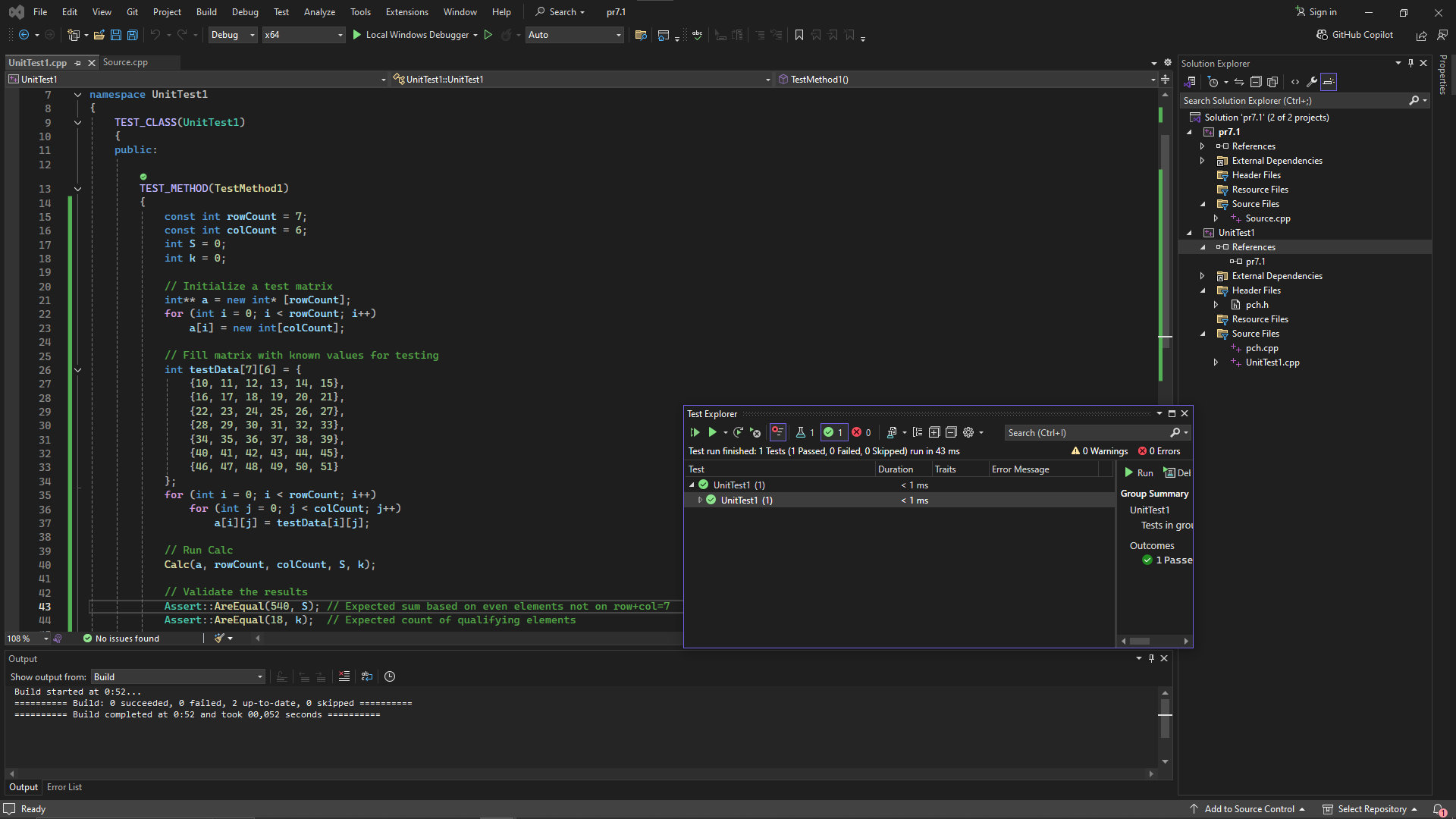
}

}

Посилання на git-репозиторій з проектом:

https://github.com/HanynDmytro/Lab7-1-itr

Результати unit-тесту



Висновки:

На цій лабараторній я навчився обчислювати кількість, суму заданих елементів матриці; а також впорядковувати матрицю, переставляючи місцями її рядки / стовпчики.