ЗВІТ

про виконання лабораторної роботи № <7.3>

*«****Опрацювання динамічних багатовимірних масивів****»*

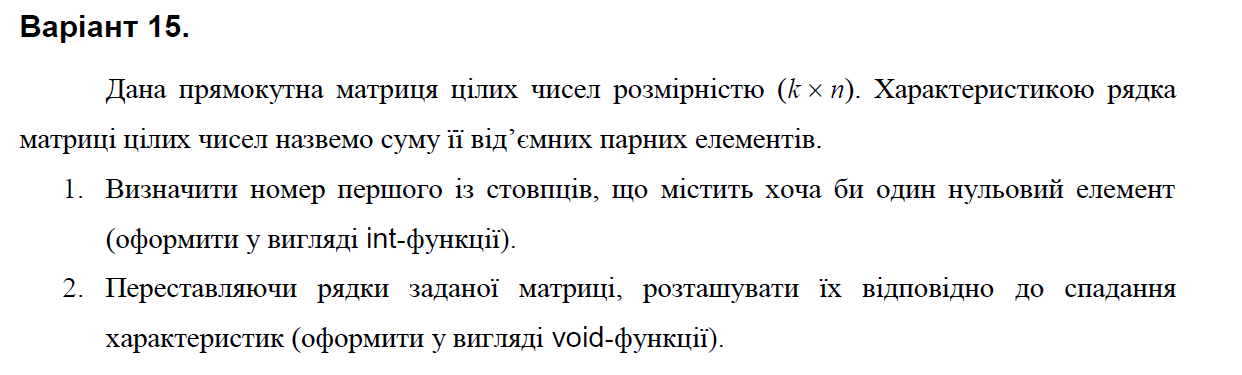
з дисципліни

«Алгоритмізація та програмування»

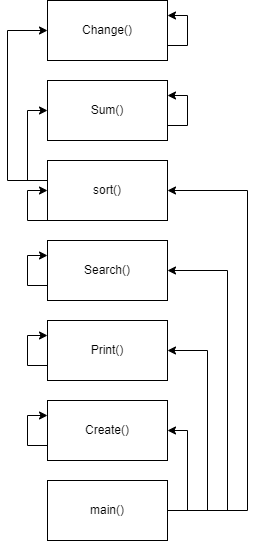
студента групи ІК-11

< *Снігура Стефана Андрійовича* >

**Умова завдання:**

****

**Структурна схема програми :**



**Текст програми:**

#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, int i, int j)

{

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

if (j < colCount - 1)

Create(a, rowCount, colCount, Low, High, i, j + 1);

else

if (i < rowCount - 1)

Create(a, rowCount, colCount, Low, High, i + 1, 0);

}

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

{

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

if (j < colCount - 1)

Print(a, rowCount, colCount, i, j + 1);

else

if (i < rowCount - 1)

{

cout << endl;

Print(a, rowCount, colCount, i + 1, 0);

}

else

cout << endl << endl;

}

int Search(int\*\* a, const int rowCount, const int colCount, int i, int j, int& k)

{

if (i < rowCount) {

if (j < colCount) {

if (a[i][j] == 0)

k = j;

if (k > 1)

return 0;

Search(a, rowCount, colCount, i, j + 1, k);

}

Search(a, rowCount, colCount, i + 1, j, k);

}

return 0;

}

int Sum(int\*\* a, int rowCount, int colCount, int j, int sumRaw) {

if (a[rowCount][j] < 0 && a[rowCount][j] % 2 == 0)

sumRaw += a[rowCount][j];

if (j < colCount - 1)

return Sum(a, rowCount, colCount, j + 1, sumRaw );

else

return sumRaw;

}

void Change(int\*\* a, const int raw1, const int raw2, const int colCount, int j)

{

int tmp;

tmp = a[raw1][j];

a[raw1][j] = a[raw2][j];

a[raw2][j] = tmp;

if (j < colCount - 1)

Change(a, raw1, raw2, colCount, j + 1);

}

void sort(int\*\* a, const int rowCount, const int colCount, int i0, int i1)

{

if (Sum(a, i1, colCount, 0, 0) < Sum(a, i1 + 1, colCount, 0, 0))

Change(a, i1, i1 + 1, colCount, 0);

if (i1 < rowCount - i0 - 2)

{

sort(a, rowCount, colCount, i0, i1 + 1);

}

else

if (i0 < rowCount - 2)

sort(a, rowCount, colCount, i0 + 1, 0);

}

int main()

{

srand((unsigned)time(NULL));

int Low = -20;

int High = 26;

int rowCount = 4;

int colCount = 4;

int k = -1;

int S = 0;

int tmp = 0;

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

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

a[i] = new int[colCount];

Create(a, rowCount, colCount, Low, High, 0, 0);

Print(a, rowCount, colCount, 0, 0);

Search(a, rowCount, colCount, 0, 0, k);

if (k < 0) {

cout << "There is any null"<< endl;

}

else

cout << "Null is in " << k + 1 << " column"<< endl;

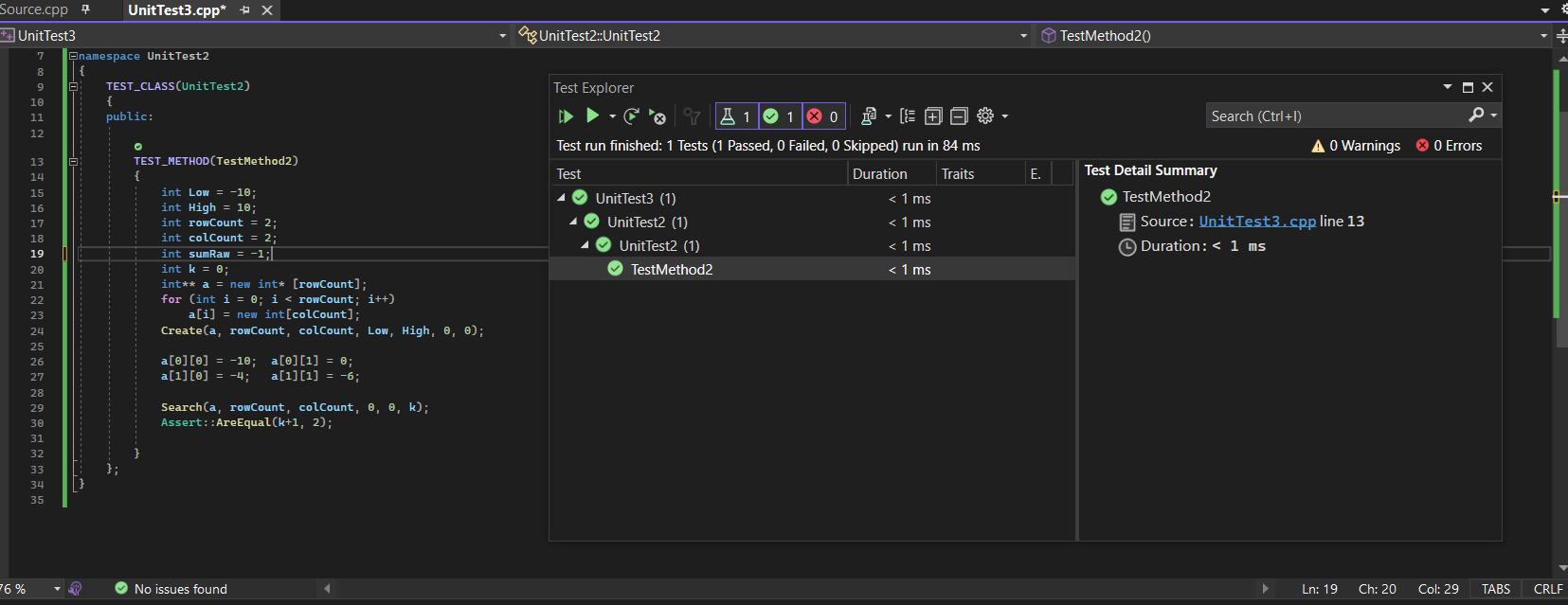
sort(a, rowCount, colCount,0,0);

Print(a, rowCount, colCount, 0, 0);

return 0;

}

**UNIT-test:**

****

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

<https://github.com/BigTrouble-Git/ashtray.git>

**Висновки**:

Виконавши цю лабораторну роботу я навчився опрацьовувати динамічні двовимірні та багатовимірні масиви.