Національний університет «Львівська політехніка»

Інститут комп’ютерних наук та інформаційних технологій



Звіт

про виконання лабораторної роботи №7.3

Опрацювання динамічних багатовимірних масивів

з дисципліни «Алгоритмізація та програмування»

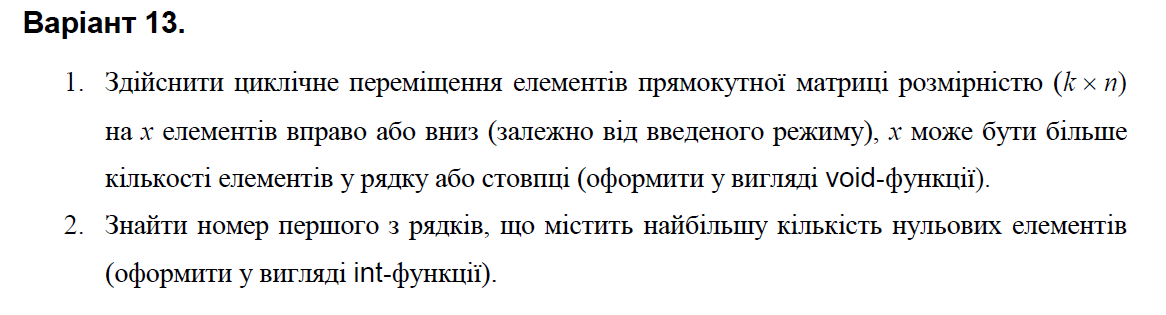
студента IT-11 Лісничука Арсена

Прийняв доцент Григорович В.Г.

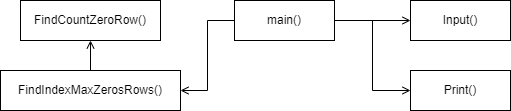
2021

Мета: Навчитися опрацьовувати динамічні двовимірні та багатовимірні масиви.

Умова завдання:



Структурна схема програми:



Текст програми:

// Main.cpp

// Лісничук Арсен

// Лабораторна робота № 7.3.1

// Опрацювання динамічних багатовимірних масивів

// Варіант 13(Ітераційний спосіб)

#include <iostream>

#include <iomanip>

using namespace std;

void Input(int\*\* arr, const int rowCount, const int colCount);

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

void Move(int\*\* arr, const int rowCount, const int colCount, int mode);

// mode - напрям: 1 - вправо, 2 - вниз

// steps - крок(на скільки елементів перемістити)

void Moves(int\*\* arr, const int rowCount, const int colCount, const int mode, const int steps);

int FindCountZeroRow(int\*\* arr, const int i, const int colCount);

int FindIndexMaxZerosRows(int\*\* arr, const int rowCount, const int colCount);

int main()

{

srand((unsigned)time(NULL));

int n, k;

cout << "k(width) = "; cin >> k;

cout << "n(height) = "; cin >> n;

int\*\* arr = new int\* [n];

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

arr[i] = new int[k];

Input(arr, n, k);

Print(arr, n, k);

while (true) {

int steps, mode;

cout << "mode(1=down, 2=right, ?=down, 0=exit) = "; cin >> mode;

if (mode == 0) break;

cout << "steps = "; cin >> steps;

Moves(arr, n, k, mode, steps);

Print(arr, n, k);

}

int index\_max\_null = FindIndexMaxZerosRows(arr, n, k);

if (index\_max\_null != -1)

cout << "index row with maximum count of zeros = " << index\_max\_null << endl;

else

cout << "there are no zeros " << endl;

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

delete[] arr[i];

delete[] arr;

return 0;

}

void Input(int\*\* arr, const int rowCount, const int colCount)

{

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

{

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

{

cout << "arr[" << i << "][" << j << "] = ";

cin >> arr[i][j];

}

cout << endl;

}

}

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

{

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

{

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

{

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

}

cout << endl;

}

cout << endl;

}

void Move(int\*\* arr, const int rowCount, const int colCount, int mode)

{

if (mode != 1 && mode != 2)

mode = 1;

if (mode == 1)

{

int\* tmp = new int[colCount];

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

tmp[j] = arr[0][j];

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

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

{

if (i == rowCount - 1)

arr[0][j] = tmp[j];

else {

int tmp1 = arr[i + 1][j];

arr[i + 1][j] = tmp[j];

tmp[j] = tmp1;

}

}

delete[] tmp;

}

if (mode == 2)

{

int\* tmp = new int[rowCount];

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

tmp[i] = arr[i][0];

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

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

{

if (j == colCount - 1)

arr[i][0] = tmp[i];

else {

int tmp1 = arr[i][j+1];

arr[i][j+1] = tmp[i];

tmp[i] = tmp1;

}

}

delete[] tmp;

}

}

void Moves(int\*\* arr, const int rowCount, const int colCount, const int mode, const int steps)

{

for (int s = 0; s < steps; s++)

Move(arr, rowCount, colCount, mode);

}

int FindCountZeroRow(int\*\*arr, const int i, const int colCount)

{

int C = 0;

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

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

C++;

return C;

}

int FindIndexMaxZerosRows(int\*\* arr, const int rowCount, const int colCount)

{

int max\_zeros = 0;

int index\_max\_zeros = -1;

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

{

int zeros = FindCountZeroRow(arr, i, colCount);

if (zeros > max\_zeros) {

max\_zeros = zeros;

index\_max\_zeros = i;

}

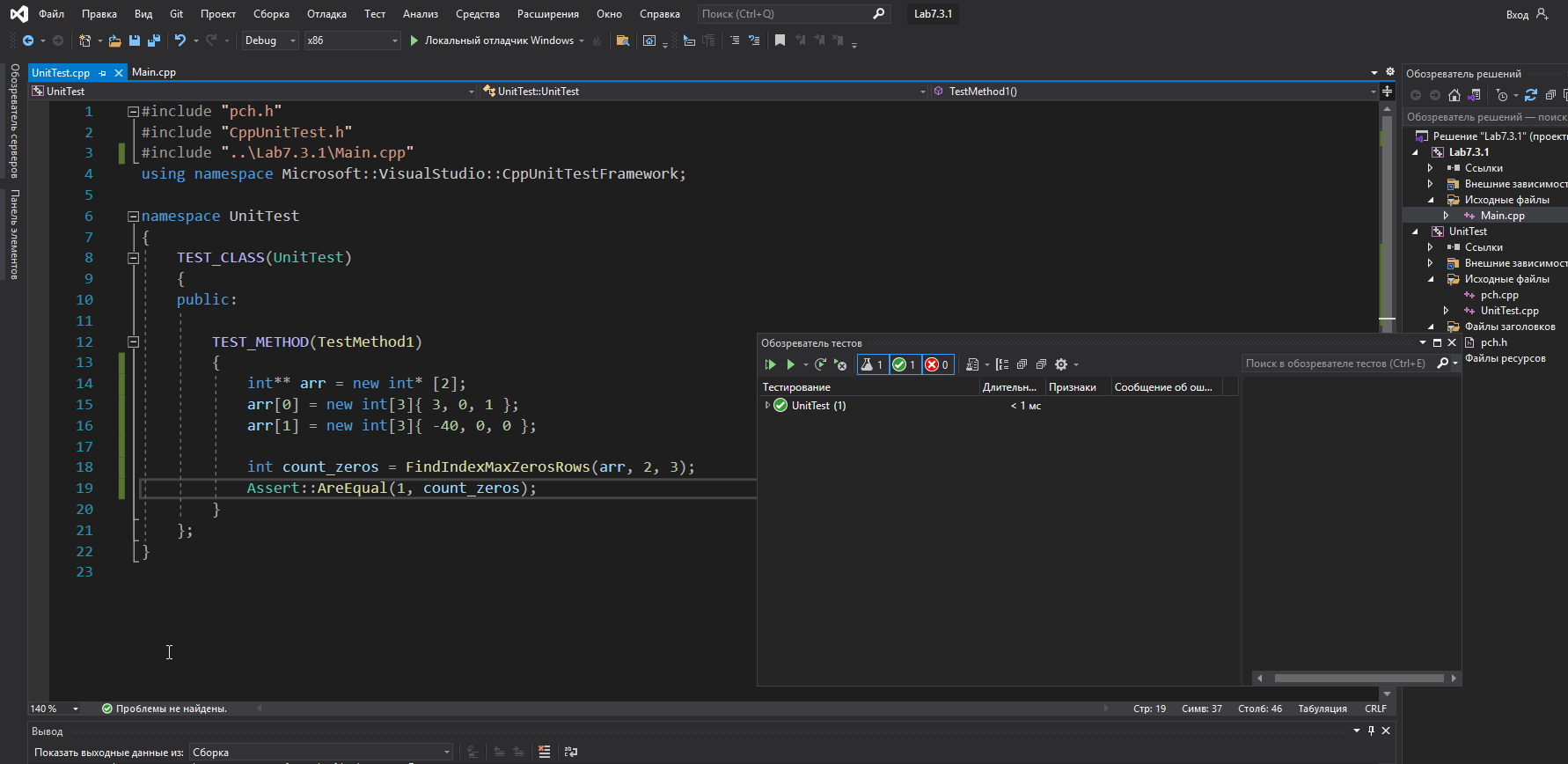
}

return index\_max\_zeros;

}}

Git-посилання: <https://github.com/fhctyw/Lab7.3.git>

Юніт тест:



Висновок: В цій лабораторній я навчитися опрацьовувати динамічні двовимірні та багатовимірні масиви.