ЗВІТ

про виконання лабораторної роботи № 7.2

«Опрацювання багатовимірних масивів ітераційним і рекурсивним способами»

з дисципліни

«Алгоритмізація та програмування»

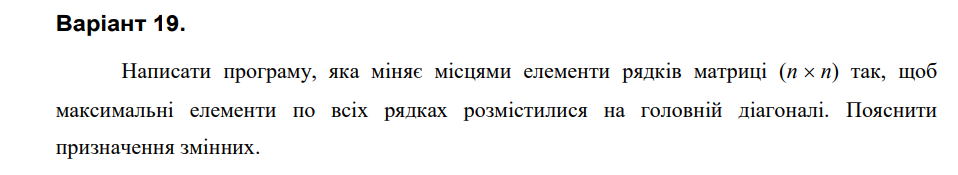
студентки групи РІ-11

Пятницької Вікторії Володимирівни

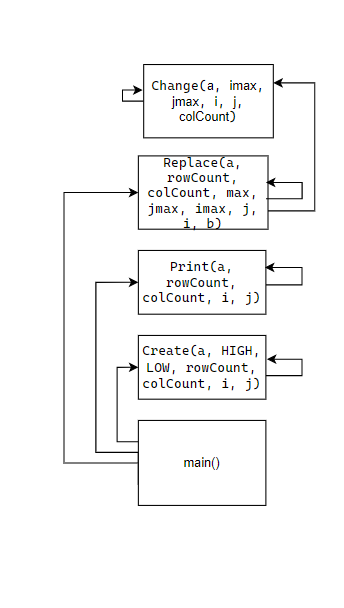
**Мета роботи**

Навчитися опрацьовувати двовимірні та багатовимірні масиви ітераційним і рекурсивним способами.

**Умова завдання:**

****

**Структурна схема програми:**

****

**Текст програми:**

#include <iostream>

#include <iomanip>

#include <time.h>

using namespace std;

void Create(int\*\* a, const int HIGH, const int LOW, const int rowCount, const int colCount, int i, int j)

{

{

a[i][j] = LOW + rand() % (HIGH - LOW + 1);

if (j < colCount - 1)

Create(a, HIGH, LOW, rowCount, colCount, i, j + 1);

else

if (i < rowCount - 1)

Create(a, HIGH, LOW, rowCount, colCount, i + 1, 0);

}

}

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

{

cout << a[i][j] << "\t";

if (j < colCount - 1)

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

else

if (i < rowCount - 1)

{

cout << "\n";

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

}

else

cout << "\n" << "\n" << "\n";

}

void Change(int\*\* a, int& imax, int& jmax, int& i, int& j, const int colCount)

{

int tmp = 0;

tmp = a[imax][jmax];

a[imax][jmax] = a[i][i];

a[i][i] = tmp;

if (j < colCount - 1)

Change(a, imax, jmax, i, j, colCount);

}

int Max(int\*\* a, const int rowCount, const int colCount, int& max, int& imax\_1, int& jmax\_1, int i, int j)

{

if (a[i][j] >= max)

{

max = a[i][j];

jmax\_1 = j;

imax\_1 = i;

}

if (j < colCount - 1)

Max(a, rowCount, colCount, max, imax\_1, jmax\_1, i, j + 1);

else

if (i < rowCount - 1)

Max(a, rowCount, colCount, max, imax\_1, jmax\_1, i + 1, 0);

else

return max;

}

void Replace(int\*\* a, const int rowCount, const int colCount, int max, int& jmax, int& imax, int j, int i, bool b)

{

if (i < rowCount)

{

if (b)

{

max = a[i][0];

b = false;

}

if (i < rowCount)

{

if (j < colCount)

{

if (a[i][j] >= max)

{

max = a[i][j];

jmax = j;

imax = i;

}

if (j == colCount - 1)

{

cout << "a[" << imax << "][" << jmax << "] = " << max << "\t" << "\n";

if (a[i][i] != a[imax][jmax])

{

Change(a, imax, jmax, i, j, colCount);

}

}

Replace(a, rowCount, colCount, max, jmax, imax, j + 1, i, b);

}

else

{

bool b = true;

Replace(a, rowCount, colCount, max, jmax, imax, 0, i + 1, b);

}

}

}

else return;

}

int main()

{

srand(time(NULL));

const int HIGH = 100;

const int LOW = 1;

int rowCount;

int colCount;

cout << "Enter the matrix size: "; cin >> rowCount;

colCount = rowCount;

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

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

a[i] = new int[colCount];

int i = 0;

int j = 0;

Create(a, HIGH, LOW, rowCount, colCount, i, j);

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

int max = a[0][0];

int imax = 0;

int jmax = 0;

bool b = false;

Replace(a, rowCount, colCount, max, jmax, imax, j, i, b);

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

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

delete[] a[i];

delete[]a;

system("pause");

return 0;

}

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

[**https://github.com/viksi01/lab\_7.2.rec**](https://github.com/viksi01/lab_7.2.rec)

**Результати unit-тесту:**

#include "pch.h"

#include "CppUnitTest.h"

#include "../lab\_7.2.rec/lab\_7.2.rec.cpp"

using namespace Microsoft::VisualStudio::CppUnitTestFramework;

namespace UnitTest72rec

{

TEST\_CLASS(UnitTest72rec)

{

public:

TEST\_METHOD(TestMethod1)

{

int result;

int imax\_1 = 0;

int jmax\_1 = 0;

int a[3][3] = {

{1, 5, 8},

{2, 15, 7},

{3, 7, 9},

};

int max = a[0][0];

int\* pa[3] = { a[0], a[1], a[2] };

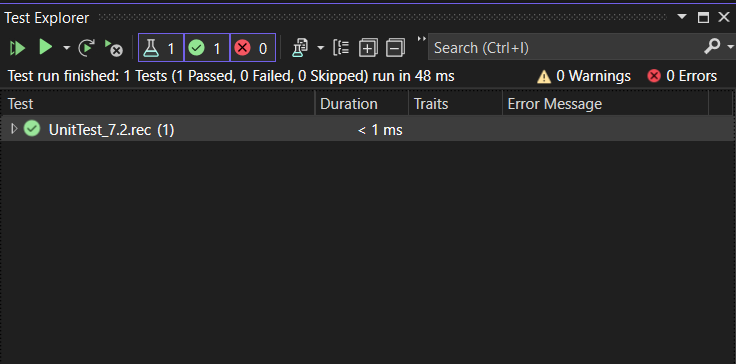
result = Max(pa, 3, 3, max, imax\_1, jmax\_1, 0 ,0); ;

Assert::AreEqual(result, 15);

}

};

}

****

**Висновки:**

Отже, виконавши цю лабораторну роботу я навчилася опрацьовувати двовимірні та багатовимірні масиви ітераційним і рекурсивним способами