МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ УКРАИНЫ

ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Факультет КНТ

Кафедра КИ

Лабораторная работа №2

по курсу «ООП»

по теме «Двумерный массив (матрица)»

Выполнил:

ст. гр. КН-16

\_\_\_\_\_\_\_\_\_\_\_ Семеренко В.К.

Проверила:

\_\_\_\_\_\_\_\_\_\_\_\_ Дикова Ю. Л.

Покровск 2017

Цель:

Освоение основ технологии объектно-ориентированного программирования на примере создания класса «матрица» для описания двумерного массива

Задание:

Обнулить элементы k-ой строки и l-го столбца прямоугольной матрицы, после чего присвоить этим элементам такие значения, что бы сумма элементов каждой строки и сумма элементов каждого столбца, за исключением l-го, была равна нулю. Значения k и l ввести с клавиатуры. Для прямоугольной матрицы найти минимальный из положительных элементов и максимальный из отрицательных элементов, после чего обменять их местами. Нулевые элементы не учитывать. найти максимальный по модулю элемент и заменить его средним арифметическим значением остальных элементов данной строки.

Файл Matrix.h:

#pragma once

class MyMatrix

{

public:

MyMatrix();

MyMatrix(int, int);

void SetMatrix(int, int);

void NullElement();

void ReverseMinMax();

void MaxModule();

void PrintMatrix();

~MyMatrix();

private:

double\*\* X;

int i\_index, j\_index;

};

Файл Matrix.cpp:

#include <iostream>

#include "Matrix.h"

using namespace std;

MyMatrix::MyMatrix() {

i\_index = 0;

j\_index = 0;

}

MyMatrix::MyMatrix(int size\_i, int size\_j) {

if (size\_i < 0 || size\_j < 0) {

cout << "You enter incorrect size. Pls enter correct" << endl;

cin >> size\_i >> size\_j;

MyMatrix(size\_i, size\_j);

}

else {

i\_index = size\_i;

j\_index = size\_j;

SetMatrix(i\_index, j\_index);

}

}

void MyMatrix::SetMatrix(int size\_i, int size\_j) {

X = new double\* [size\_i];

for (int i = 0; i < size\_i; i++)

X[i] = new double[size\_j];

cout << "Enter Matrix: " << endl;

for (int i = 0; i < size\_i; i++)

for (int j = 0; j < size\_j; j++)

cin >> X[i][j];

}

void MyMatrix::NullElement() {

int k, l;

cout << "Enter stroke" << endl;

cin >> k;

cout << "Enter colomn" << endl;

cin >> l;

for (int i = 0; i < i\_index; i++) {

if (i == k) {

for (int j = 0; j < j\_index; j++) {

if (j == l) {

X[i][j] = 0;

}

}

}

}

}

void MyMatrix::ReverseMinMax() {

double min = X[0][0], max = 0;

int index\_min\_i = 0, index\_min\_j = 0, index\_max\_i, index\_max\_j;

for (int i = 0; i < i\_index; i++)

for (int j = 0; j < j\_index; j++)

if (X[i][j] < 0) {

max = X[i][j];

index\_max\_i = i;

index\_max\_j = j;

break;

}

for(int i=1;i<i\_index;i++)

for (int j = 0; j < j\_index; j++) {

if (X[i][j] < min && X[i][j]>0) {

min = X[i][j];

index\_min\_i = i;

index\_min\_j = 0;

}

if (X[i][j] > max && X[i][j] < 0) {

index\_max\_i = i;

index\_max\_j = j;

}

}

double tmp;

tmp = X[index\_max\_i][index\_max\_j];

X[index\_max\_i][index\_max\_j] = X[index\_min\_i][index\_min\_j];

X[index\_min\_i][index\_min\_j] = tmp;

}

void MyMatrix::MaxModule() {

double ModuleMax = abs(X[0][0]);

double MiddleCount=0;

int ModuleMax\_i, ModuleMax\_j;

for (int i = 0; i < i\_index; i++)

for (int j = 0; j < j\_index; j++)

if (abs(X[i][j]) > ModuleMax) {

ModuleMax = abs(X[i][j]);

ModuleMax\_i = i;

ModuleMax\_j = j;

}

for (int i = 0; i < i\_index; i++) {

if (i == ModuleMax\_i) {

for (int j = 0; j < j\_index; j++)

MiddleCount += X[i][j];

}

}

MiddleCount /= j\_index;

X[ModuleMax\_i][ModuleMax\_j] = MiddleCount;

}

void MyMatrix::PrintMatrix() {

for (int i = 0; i < i\_index; i++) {

for (int j = 0; j < j\_index; j++) {

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

}

cout << endl;

}

}

MyMatrix::~MyMatrix() {

delete X;

}

Файл Main.cpp:

#include <math.h>

#include <stdio.h>

#include <stdlib.h>

#include <iostream>

#include "Matrix.h"

using namespace std;

int main() {

MyMatrix Matrix(5, 2);

Matrix.PrintMatrix();

Matrix.NullElement();

Matrix.PrintMatrix();

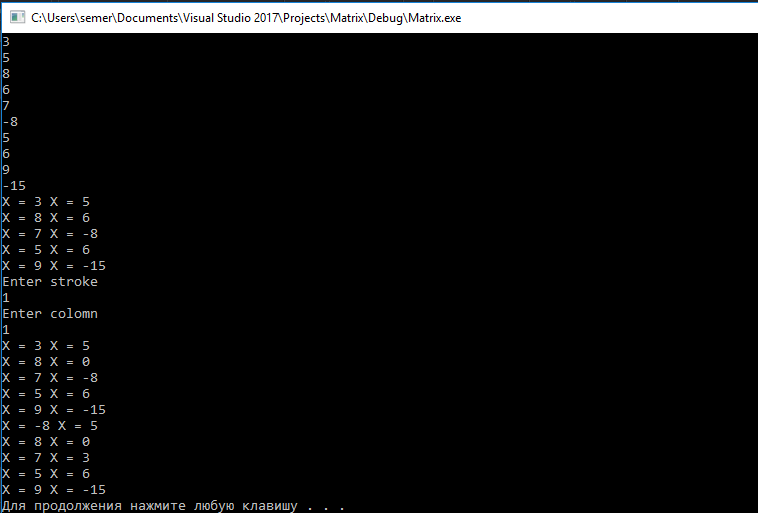
Matrix.ReverseMinMax();

Matrix.PrintMatrix();

system("pause");

}

Листинг:



Вывод:

В ходе выполнения лабораторной работы я освоил основы технологии объектно-ориентированного программирования на примере создания класса «матрица» для описания двумерного массива