**Лабораторная работа № 23**

**Тема:** **Программирование с использованием шаблона функции.**

**Задание:** Дана прямоугольная матрица. Определить номер первого из столбцов, содержащих хотя бы один нулевой элемент.

Характеристикой строки целочисленной матрицы назовем сумму

ее отрицательных элементов. Переставляя строки заданной матрицы, расположить их в соответствии c убыванием характеристик.

**Текст программы:**

#include <iostream>

#include <cstdlib>

#include <ctime>

#include <typeinfo>

using namespace std;

template <class T>

void IsNull(T\*\* array, int rows, int columns)

{

bool null\_col = true;

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

{

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

{

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

{

cout << "индекс первого столбца, содержащего нулевой элемент = " << i << endl;

null\_col = false;

i = columns;

break;

}

}

}

if (null\_col)

cout << "в матрице отсутствуют нулевые элементы " << endl;

}

template <class T>

void NewArray(T\*\* array, int rows, int columns)

{

int j = 1;

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

{

for (j; j < columns; j++)

{

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

cin >> array[i][j];

}

j = 0;

}//заполнение исходной матрицы

}

int\*\* CreateArray(int rows, int columns, int first)

{

int\*\* array = new int\*[rows];

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

array[i] = new int[columns];

array[0][0] = first;

return array;//создание матрицы

}

double\*\* CreateArray(int rows, int columns, double first)

{

double\*\* array = new double\*[rows];

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

array[i] = new double[columns];

array[0][0] = first;

return array;//создание матрицы

}

template <class T>

void Charact(T\*\* array, T\* Charact, int rows, int columns)

{

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

{

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

{

if (array[i][j] < 0)

{

Charact[i] += array[i][j];

}

}

cout << Charact[i] << " " << endl;

}

}

template <class T>

void Sort(T\*\* array, T\* Charact, int rows, int columns)

{

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

{

T max = Charact[i];

int max\_ind = i;

for (int k = (i+1); k < rows; k++)

{

if (Charact[k] > max)

{

max = Charact[k];

max\_ind = k;

}

}

T\* temp = array[i];

array[i] = array[max\_ind];

array[max\_ind] = temp;

int temp\_char = Charact[i];

Charact[i] = Charact[max\_ind];

Charact[max\_ind] = temp\_char;

}

}

template <class T>

void Print(T\*\* array, int rows, int columns)

{

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

{

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

{

cout << array[i][j] << " ";

}

cout << endl;

}

cout << endl;

}

int main()

{

int rows, columns;

char typeArr;

setlocale(LC\_ALL, "Russian");

srand(time(0));

cout << "Введите число столбцов ";

cin >> columns;

cout << "Введите число строк ";

cin >> rows;

cout << "Введите тип массива: int - i; double - d ";

cin >> typeArr;

switch (typeArr)

{

case 'i': {

int first;

cout << "A[0][0] = ";

cin >> first;

int\*\* array = CreateArray(rows, columns, first);

int\* charact = new int [rows] {0};

NewArray(array, rows, columns);

IsNull(array, rows, columns);

Print(array, rows, columns);

Charact(array, charact, rows, columns);

Sort(array, charact, rows, columns);

Print(array, rows, columns);

break; }

case 'd': {

double first;

cout << "A[0][0] = ";

cin >> first;

double\*\* array = CreateArray(rows, columns, first);

double\* charact = new double [rows] {0};

NewArray(array, rows, columns);

IsNull(array, rows, columns);

Print(array, rows, columns);

Charact(array, charact, rows, columns);

Sort(array, charact, rows, columns);

Print(array, rows, columns);

break; }

}

}

**Блок-схема:**

****

