Белорусский государственный технологический университет

Факультет информационных технологий

Кафедра программной инженерии

 Лабораторная работа 14

По дисциплине «Основы алгоритмизации и программирования»

На тему «Многомерные массивы»

Выполнила:

Студент(ка) 1 курса 7 группы

Подшиваленко Диана Игоревна

Преподаватель: асс. Андронова М.В.

2023, Минск

|  |  |
| --- | --- |
| **11** | 1. Найти наибольший элемент главной диагонали матрицы **A(n, n)** и вывести на печать всю строку, в которой он находится.  2. Определить в матрице первый столбец, все элементы которого отрицательны, и среднее арифметическое этих элементов. Вычесть полученное значение из всех элементов матрицы. |

#include <iostream>

#include <iomanip>

#include <Windows.h>

#include <stdio.h>

#include <conio.h>

#include <vector>

#include <string>

using namespace std;

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int n;

int a[500][500]; // создаём двумерный массив

cout << "Введите размер массива ";

cin >> n; //считываем размер массива

cout << "Введите элементы массива " << endl;

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

for (int j = 0; j < n; j++) {

cin >> a[i][j]; // считываем элементы массива

}

}

int max = a[0][0]; // присваиваем переменной max значение 1 элемента главной диагонали

int num = 0; // переменная для хранения номера строки

for (int i = 1; i < n; i++) { // ищем максимальный элемент главной диагонали

if (a[i][i] > max) {

max = a[i][i];

num = i;

}

}

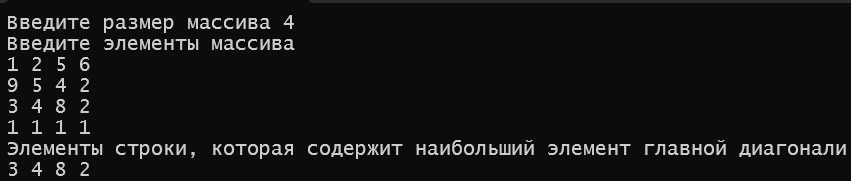
cout << "Элементы строки, которая содержит наибольший элемент главной диагонали " << endl;

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

cout << a[num][i] << ' '; // выводим строку с максимальным элементом главной диагонали

}

}



#include <iostream>

#include <iomanip>

#include <Windows.h>

#include <stdio.h>

#include <conio.h>

#include <vector>

#include <string>

using namespace std;

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int n, m, num = 0, s = 0;

bool t = false;

int a[500][500]; // создаём двумерный массив

cout << "Введите размер массива (количество строк и столбцов)";

cin >> n >> m; //считываем размер массива (у нас же не обязательно квадратная матрица)

cout << "Введите элементы массива " << endl;

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

for (int j = 0; j < m; j++) {

cin >> \*(\*(a + i) + j); // считываем элементы массива

}

}

for (int i = 0; i < m; i++) { // ищем столбец с отрицательными элементами

for (int j = 0; j < n; j++) {

if (\*(\*(a + j) + i) >= 0) {

break;

}

else {

s += 1;

}

}

if (s == n) {

t = true;

num = i;

break;

}

s = 0;

}

s = 0;

if (t == false) { // если не нашли, то выводим сообщение об этом

cout << "Такого столбца нет";

}

else if (t == true) { // если нашли

cout << "Это столбец с номером " << num + 1 << endl; // это если считать что первый столбец имеет номер 1

for (int i = 0; i < n; i++) { //ищем среднее арифметическое

s += \*(\*(a + i) + num);

}

s /= n;

cout << "Среднее арифметическое равно " << s << endl;

for (int i = 0; i < n; i++) { // вычитаем среднее арифметическое из каждого элемента массива

for (int j = 0; j < m; j++) {

\*(\*(a + i) + j) -= s;

}

}

cout << "Полученный массив" << endl;

for (int i = 0; i < n; i++) { // выводим массив

for (int j = 0; j < m; j++) {

cout << \*(\*(a + i) + j) << ' ';

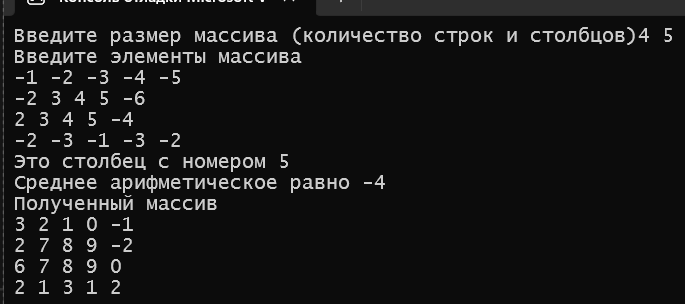
}

cout << endl;

}

}

}



**6. Дополнительные задания.**

**1. Дана квадратная матрица порядка 2n, элементы которой фо рмируются случайным образом и находятся в пределах от −10 до 10. Получить новую матрицу, переставляя ее блоки размера n×n в соответствии со схемой.**

#include <iostream>

#include <iomanip>

#include <Windows.h>

#include <stdio.h>

#include <conio.h>

#include <vector>

#include <string>

using namespace std;

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int n, m;

bool t = false;

int a[500][500]; // создаём двумерный массив

cout << "Введите размер массива (количество строк и столбцов)";

cin >> n; //считываем размер массива

m = 2 \* n;

srand(time(NULL)); // чтобы каждый раз генерировались случайные числа

for (int i = 0; i < m; i++) { // заполняем матрицу случайными числами в диапазоне [-10;10]

for (int j = 0; j < m; j++) {

a[i][j] = -10 + rand() % 21;

}

}

cout << "Исходная матрица " << endl;

for (int i = 0; i < m; i++) { // выводим исходную матрицу

for (int j = 0; j < m; j++) {

cout << setw(3) << a[i][j] << " "; // для красоты вывода задаём ширину вывода под каждый элемент = 3

}

cout << endl;

}

cout << endl;

int s; // создаём буферную переменную

for (int i = 0; i < n; i++) { // меняем левый верхний блок с правым нижним

for (int j = 0; j < n; j++) {

s = a[i][j];

a[i][j] = a[i + n][j + n];

a[i + n][j + n] = s;

}

}

for (int i = n; i < m; i++) { // меняем левый нижний блок с правым верхним

for (int j = 0; j < n; j++) {

s = a[i][j];

a[i][j] = a[i - n][j + n];

a[i - n][j + n] = s;

}

}

cout << "Полученная матрица " << endl;

for (int i = 0; i < m; i++) { // выводим полученную матрицу

for (int j = 0; j < m; j++) {

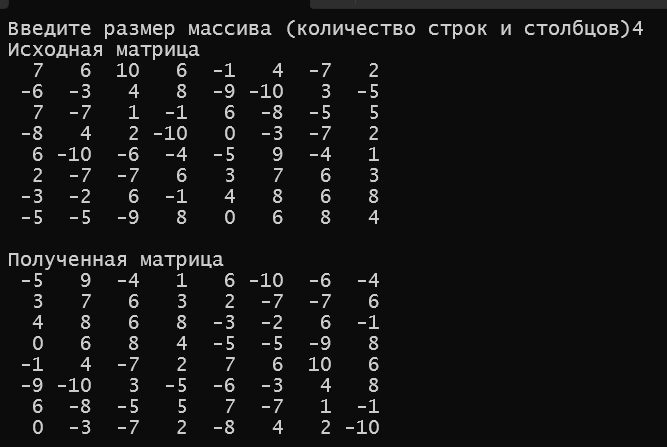
cout << setw(3) << a[i][j] << " "; // для красоты вывода задаём ширину вывода под каждый элемент = 3

}

cout << endl;

}

}



**2. Латинским квадратом порядка n называется квадратная таблица размером nхn, каждая строка и каждый столбец которой содержат все числа от 1 до n. Для заданного n в матрице L(n, n) построить латинский квадрат порядка n.**

#include <iostream>

#include <iomanip>

#include <Windows.h>

#include <stdio.h>

#include <conio.h>

#include <vector>

#include <string>

using namespace std;

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int n, m;

bool t = false;

int a[500][500]; // создаём двумерный массив

cout << "Введите размер массива (количество строк и столбцов)";

cin >> n; //считываем размер массива

for (int i = 0; i < n; i++) { // формируем латинский квадрат

m = i + 1;

for (int j = 0; j < n; j++) {

a[i][j] = m;

m += 1;

if (m > n) {

m = 1;

}

}

}

for (int i = 0; i < n; i++) { // выводим матрицу на экран

for (int j = 0; j < n; j++) {

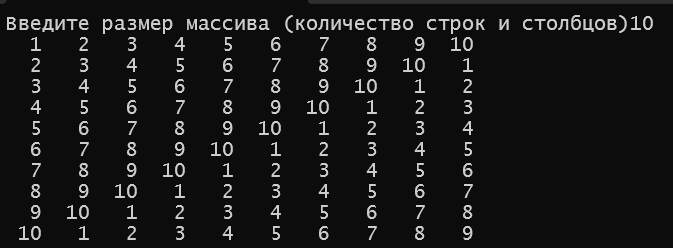
cout << setw(3) << a[i][j] << " ";

}

cout << endl;

}

}

****

**3. Путем перестановки элементов квадратной вещественной матрицы добиться того, чтобы ее максимальный элемент находился в левом верхнем углу, следующий по величине − в позиции (2, 2), следующий − в позиции (3, 3) и т. д., заполнив таким образом всю главную диагональ.**

#include <iostream>

#include <iomanip>

#include <Windows.h>

#include <stdio.h>

#include <conio.h>

#include <vector>

#include <string>

using namespace std;

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int n;

double t; //буферная переменная

double a[200][200]; // создаём двумерный массив

double x[400] = {0};

int k = 0;

cout << "Введите размер массива (количество строк и столбцов)";

cin >> n; //считываем размер массива

for (int i = 0; i < n; i++) { // считываем элементы матрицы и + записываем их в отдельный массив для сортировки

for (int j = 0; j < n; j++) {

cin >> a[i][j];

x[k] = a[i][j];

k++;

}

}

for (int i = 0; i < k; i++) { //сортируем массив для главной диагонали

for (int j = (k - 1); j >= (i + 1); j--) {

if (x[j] < x[j - 1]) {

t = x[j];

x[j] = x[j - 1];

x[j - 1] = t;

}

}

}

k--;

for (int i = 0; i < n; i++) { // меняем элементы главной диагонали

a[i][i] = x[k];

k--;

}

for (int i = 0; i < n; i++) { // выводим матрицу на экран

for (int j = 0; j < n; j++) {

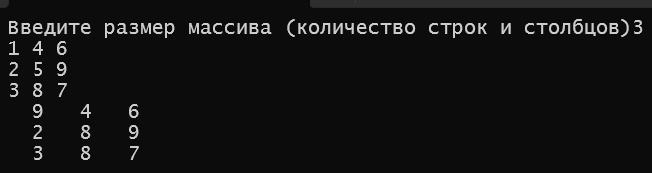
cout << setw(3) << a[i][j] << " ";

}

cout << endl;

}

}

****