Белорусский государственный технологический университет

Факультет информационных технологий

Кафедра программной инженерии

 Лабораторная работа 14

По дисциплине «Основы алгоритмизации и программирования»

На тему «Многомерные массивы»

Выполнил:

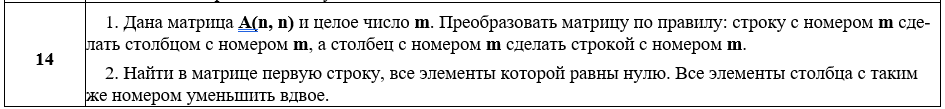
Студент 1 курса 10 группы

Сегренёв Кирилл Сергеевич

Преподаватель: асс. Андронова М.В.

2023, Минск

**ВАРИАНТ 14**



**Задание 1.**

**Код:**

#include <iostream>

#include <vector>

using namespace std;

int main() {

int m;

cout << "input m: ";

cin >> m;

const int n = 3;

int arr[n][n];

if (m >= 1 && m <= n) { // Если m входит в диапазон размерности массива, то продолжать выполнения действий

cout << "input array: " << endl;

for (int i = 0; i < n; i++) { // Вводим элементы массива

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

cin >> arr[i][g];

}

}

cout << '\n';

for (int i = 0; i < n; i++) { // Меняем строку и столбец местами меняя индексы

int temp = arr[i][m - 1];

arr[i][m - 1] = arr[m - 1][i];

arr[m - 1][i] = temp;

}

cout << "changed array: " << endl;

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

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

cout << arr[i][g] << ' ';

}

cout << '\n';

}

}

else { // Ошибка, если m не входит в диапазон длины массива

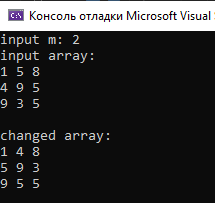
cout << "ERROR" << endl;

}

return 0;

}

**Output:**



**Задание 2.**

**Код:**

#include <iostream>

using namespace std;

void main() {

// Инициализируем массив

int arr[4][4] = {

{3, 10, 6, 19},

{0, 0, 0, 0},

{7, 4, 2, 1},

{34, 62, 10, 4}

};

int j, k;

for (int i = 0; i < 4; i++) {

k = 0;

for (int g = 0; g < 4; g++) {

if (arr[i][g] == 0) {

k++; // Индикатор для нахождения строки нулевых элементов

}

if (k == 4) {

j = i; // Присваиваем индекс строки переменной, которая в другом цикле будет являться столбцом

}

}

}

for (int i = 0; i < 4; i++) { // Делим элементы столбца j на 2

arr[i][j] /= 2;

}

for (int i = 0; i < 4; i++) { // Вывод элементов массива

for (int g = 0; g < 4; g++) {

cout << arr[i][g] << ' ';

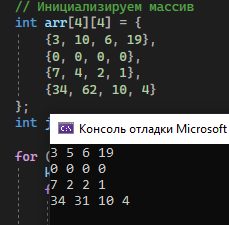
}

cout << '\n';

}

}

**Output:**



**ДОПОЛНИТЕЛЬНЫЕ ЗАДАНИЯ**

**Задание 1**

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

**Код:**

#include <iostream>

#include <cstdlib>

#include <ctime>

using namespace std;

void main() {

srand(time(0));

const int N = 2;

const int n = 2 \* N;

int arr[n][n];

for (int i = 0; i < n; i++) { // Записываем в наш массив элементы(от -10 до 10)

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

arr[i][g] = rand() % 21 - 10;

}

}

cout << "first: " << '\n';

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

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

cout << arr[i][g] << '\t';

}

cout << '\n';

}

for (int i = 0; i < N; i++) { // Меняем положение элементов согласно схеме

for (int g = 0; g < N; g++) {

int temp = arr[i][g];

arr[i][g] = arr[N + i][N + g];

arr[N + i][N + g] = temp;

int temp1 = arr[N + i][g];

arr[N + i][g] = arr[i][N + g];

arr[i][g + N] = temp1;

}

}

cout << "last: " << '\n';

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

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

cout << arr[i][g] << '\t';

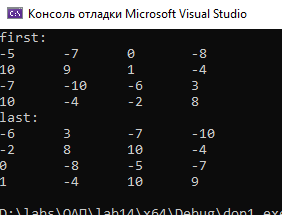
}

cout << '\n';

}

}

**Output:**



**Задание 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**.

**Код:**

#include <iostream>

using namespace std;

int main()

{

const int n = 5;

int arr[n][n];

for (int i = 0; i < n; i++){ // Создаем массив с элементами от 1 до 5

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

arr[i][g] = (i + g) % n + 1; // Присваиваем каждой диагонали число

}

}

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

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

cout << arr[i][g] << " ";

}

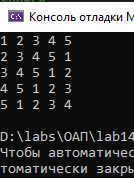
cout << '\n';

}

return 0;

}

**Output:**



**Задание 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.

**Код:**

#include <iostream>

#include <cstdlib>

#include <ctime>

#include <iomanip>

using namespace std;

int main() {

srand(time(0));

const int n = 4;

float arr[n][n];

float max = 0;

for (int i = 0; i < n; i++) { // Создаем новый массив и записываем в него элементы с помощью функции rand()

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

arr[i][g] = static\_cast<float>(rand()) / RAND\_MAX \* 20;

}

}

for (int i = 0; i < n; i++) { // В данном цикле находим максимальный элемент

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

if (max < arr[i][g]) {

max = arr[i][g];

}

}

}

for (int i = 0; i < n; i++) { // В этом цикле мы ставим максимальный элемент на главную диагональ

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

if (i == g) {

arr[i][g] = max;

}

}

}

for (int i = 0; i < n; i++) { // Выводи массив с заданной точностью

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

cout << setprecision(3) << arr[i][g] << '\t';

}

cout << '\n';

}

}

**Output:**

