Беларускі дзяржаўны тэхналагічны ўніверсітэт

Факультэт інфармацыйных тэхналогій

Кафедра праграмнай інжэнерыі

Лабараторная работа 14

Па дысцыпліне «Асновы алгарытмізацыі і праграмавання»

На тэму «Шматмерныя масівы»

Выканала:

Студэнтка 1 курса 6 группы

Жучкевіч Кацярына Сяргееўна

Выкладчык: асс. Андронава М.В.

2023, Мінск

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.

#include <iostream>

#include <ctime>

int main()

{

setlocale(0, "");

srand(time(NULL));

using namespace std;

int A[100][100]{};

int i, j, n, m, temp;

cout << "Памер квадратнай матрыцы: ";

cin >> n;

m = 2 \* n;

for (i = 0; i < m; i++) {//цыкл, які запаўняе матрыцу рандомнымі лічбамі

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

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

}

}

cout << endl;

cout << "Масiў:" << endl;

for (i = 0; i < m; i++) {//цыкл, які выводзіць матрыцу

cout << "\n";

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

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

}

}

for (i = 0; i < n; i++) { //замена элементов левого верхнего блока с правым нижним

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

temp = A[i][j];

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

A[i + n][j + n] = temp;

}

}

for (i = n; i < m; i++) { //змяняем лічба, як патрабуе практыкаванне

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

temp = A[i][j];

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

A[i - n][j + n] = temp;

}

}

cout << endl;

cout << "Атрыманы масiў:" << endl; //вывод на экран преобразованного массива

for (i = 0; i < m; i++) {

cout << "\n";

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

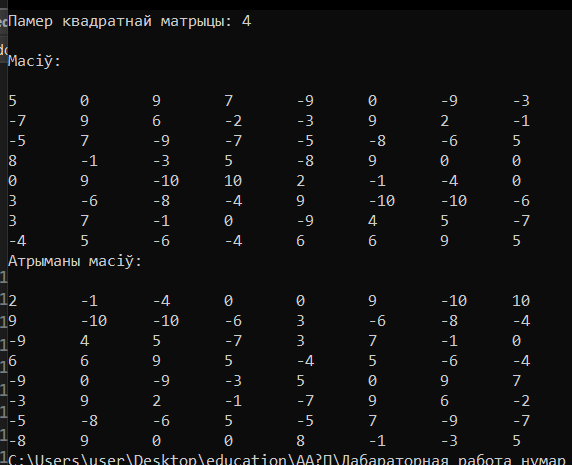
cout << A[i][j] << "\t";

}

}

return 0;

}



1. 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 <ctime>

int main()

{

setlocale(0, "");

srand(time(NULL));

using namespace std;

int L[100][100]{};

int i, j, n, temp = 1;

cout << "Памер матрыцы: ";

cin >> n;

for (i = 0; i < n; i++) { //запаўняем матрыцу

temp = i + 1;

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

L[i][j] = temp;

temp++;

if (temp > n) { //калі пераменная выходзіць за памер матрыцы...

temp = 1;//то пачынаць спачатку

}

}

}

cout << endl;

cout << "Новая матрыца:" << endl; //выводзім новую матрыцу

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

cout << "\n";

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

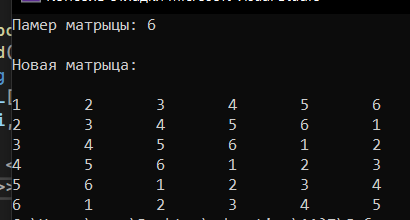
cout << L[i][j] << "\t";

}

}

return 0;

}



1. 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 <ctime>

void main()

{

setlocale(0, "");

srand(time(NULL));

using namespace std;

float A[100][100]{}, B[100]{};

int i, j, n, temp = 0, d = 0;

cout << "Памер матрыцы: ";

cin >> n;

for (i = 0; i < n; i++) { //цыкл для запаўнення матрыцы

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

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

}

}

cout << "\nМатрыца:" << endl;

for (i = 0; i < n; i++) {//выводзім масіў

cout << "\n";

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

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

}

}

for (i = 0; i < n; i++) { //запаўняем масіў, які будзе трымаць лічбы асноўнага

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

B[d] = A[i][j];

d++;

}

}

for (int i = 0; i < n - 1; i++) {//сартыруем масіў пузырком

for (int j = 0; j < n - i - 1; j++) {

if (B[j] > B[j + 1]) {

temp = B[j];

B[j] = B[j + 1];

B[j + 1] = temp;

}

}

}

d = 0;

j = 0;

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

A[i][j] = B[d];

d++;

j++;

}

cout << "Новы масiў:" << endl; //выводзім тое, што атрымалі

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

cout << "\n";

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

cout << "A[" << i << "," << j << "] = " << A[i][j] << "\t";

}

}

}

