**Бланк выполнения задания**

**Задание 3.1**

Вариант 5. Ввести с клавиатуры целые числа и сформировать из них массив a[n][n]. Написать программу, позволяющую исключать из нее строку, в которой расположен минимальный элемент главной диагонали. Преобразованный массив вывести на экран в виде матрицы.

1. Для решения поставленной задачи построены следующие функции:

void show(int a[][NMAX], int n, int m) – функция вывода матрицы;

void input(int a[][NMAX], int n) – функция ввода матрицы;

int remrow(int a[][NMAX], int n) – функция исключения из матрицы строки, в которой расположен минимальный элемент главной диагонали.

1. Код программы с использованием функций.

// Example program

#include <iostream>

#include <iomanip>

using namespace std;

#define NMAX 100

//вывод матрицы

void show(int a[][NMAX], int n, int m)

{

for (int i = 0; i<n; i++) //цикл по строкам

{

for (int j = 0; j<m; j++)//цикл по столбцам

{

cout << setw(5) << a[i][j];

}

cout << endl;

}

}

//ввод матрицы

void input(int a[][NMAX], int n)

{

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

{

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

{

cout << "a[" << i + 1 << "][" << j + 1 << "] = ";

cin >> a[i][j];

}

}

}

//функция исключения из матрицы строки, в которой расположен минимальный элемент главной диагонали.

int remrow(int a[][NMAX], int n)

{

int imin = 0;

for (int i = 1; i < n; i++)

{

if (a[i][i] < a[imin][imin])

{

imin = i;

}

}

for (int i = imin; i < n-1; i++)

{

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

{

a[i][j] = a[i + 1][j];

}

}

return imin;

}

int main()

{

int a[NMAX][NMAX];

int n;

cout << "n = ";

cin >> n;

input(a, n);

cout << "Matrix:" << endl;

show(a, n, n);

int imin=remrow(a, n);

cout << "Removed row: " << imin + 1 << endl;

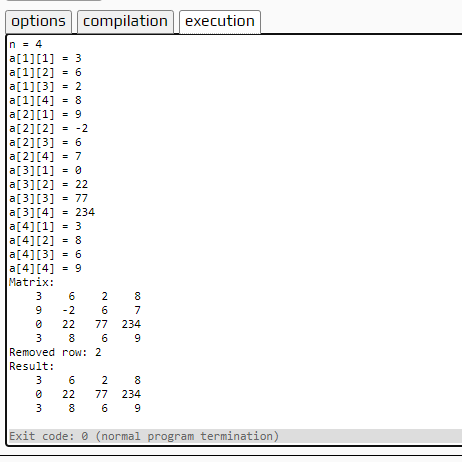
cout << "Result:" << endl;

int m = n - 1;

show(a, m, n);

}

1. Результаты вычислений (скрин результатов).



**Задание 3.2**

Вариант 5. Ввести с клавиатуры целые числа и сформировать из них массив a[n][m]. Написать программу, позволяющую исключать из нее строку и столбец, в которых расположен минимальный элемент

1. Для решения поставленной задачи построены следующие функции и прототипы функций:

void show(int a[][NMAX], int n, int m) – функция вывода матрицы;

void input(int a[][NMAX], int n, int m) – функция ввода матрицы;

void remrowcol(int a[][NMAX], int n, int m) – функция исключающая из матрицы строку и столбец, в которых расположен минимальный элемент;

1. Код программы с использованием прототипов функций.

// Example program

#include <iostream>

#include <iomanip>

using namespace std;

#define NMAX 100

//вывод матрицы

void show(int a[][NMAX], int n, int m);

//ввод матрицы

void input(int a[][NMAX], int n, int m);

// исключать из нее строку и столбец, в которых расположен минимальный элемент

void remrowcol(int a[][NMAX], int n, int m);

int main()

{

int a[NMAX][NMAX];

int n,m;

cout << "n = ";

cin >> n;

cout << "m = ";

cin >> m;

input(a, n,m);

cout << "Matrix:" << endl;

show(a, n, m);

remrowcol(a, n, m);

n--;

m--;

cout << "Result:" << endl;

show(a, n, m);

}

//вывод матрицы

void show(int a[][NMAX], int n, int m)

{

for (int i = 0; i<n; i++) //цикл по строкам

{

for (int j = 0; j<m; j++)//цикл по столбцам

{

cout << setw(5) << a[i][j];

}

cout << endl;

}

}

//ввод матрицы

void input(int a[][NMAX], int n, int m)

{

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

{

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

{

cout << "a[" << i + 1 << "][" << j + 1 << "] = ";

cin >> a[i][j];

}

}

}

//функция исключающая из матрицы строку и столбец, в которых расположен минимальный элемент

void remrowcol(int a[][NMAX], int n, int m)

{

int imin = 0, jmin = 0;

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

{

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

{

if (a[i][j] < a[imin][jmin])

{

imin = i;

jmin = j;

}

}

}

for (int i = imin; i < n - 1; i++)

{

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

{

a[i][j] = a[i + 1][j];

}

}

for (int i = jmin; i < m - 1; i++)

{

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

{

a[j][i] = a[j][i + 1];

}

}

}

1. Результаты вычислений (скрин результатов).

