Министерство образования и науки Кыргызской Республики

Кыргызский государственный технический университет

им.И.Раззакова

Факультет информационных технологий

Кафедра «Программное обеспечение компьютерных систем»

Направление:710400 «Программная инженерия»

ОТЧЕТ

По дисциплине: «Введение в программную инженерию»

Лабораторная работа №5

Тема: «Программирование Двумерных массивов»

Выполнила: студентка группы

ПИ(б)-5-19 Ажиходжоева Каныкей

Проверил: доцент Искаков Р.Т

Бишкек – 2020

1. Задан двумерный массив, содержащий 3 строки и 4 столбца. Найти наибольший элемент массива, номер строки и столбца, в которых он расположен.

* Код программы

#include <iostream>

#include <ctime>

using namespace std;

int main() {

const int N = 3, M = 4;

int a[N][M] = {};

setlocale(LC\_ALL, "RUS");

srand(time(NULL));

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

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

a[i][j] = rand() % 101;

cout << "a[" << i << "][" << j << "]=" << a[i][j] << "\t";

}

cout << endl;

}

int max=a[0][0], index1=0, index2=0;

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

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

if (a[i][j] > max) {Ч

max = a[i][j];

index1 = i;

index2 = j;

}

}

}

cout << endl;

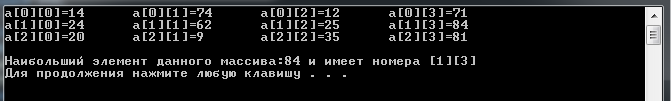
cout << "Наибольший элемент данного массива:" << max << " и имеет номера [" << index1 <<"][" <<index2 <<"]"<< endl;

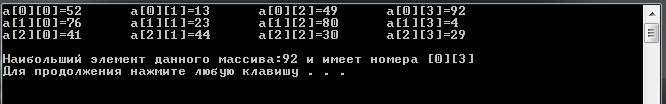
system("pause");

return 0;

}

* Результат вычислений





2.Определить количество положительных элементов каждого столбца двумерного массива, содержащего 5 строк и 5 столбцов.

* Код программы

#include <iostream>

#include<ctime>

using namespace std;

int main() {

setlocale(LC\_ALL, "RUS");

srand(time(NULL));

const int N = 5;

int a[N][N] = {}, b[N] = {},kol;

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

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

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

cout << "a[" << i << "][" << j << "]=" << a[i][j] << "\t";

}

cout << endl;

}

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

kol = 0;

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

if (a[i][j] > 0) {

kol++;

}

}

b[j] = kol;

}

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

cout << "Количество положительных элементов " << i << "-столбца " << b[i] << endl;

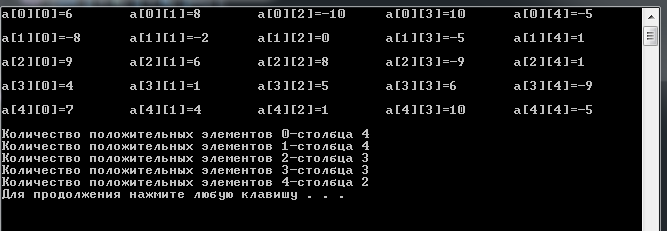
}

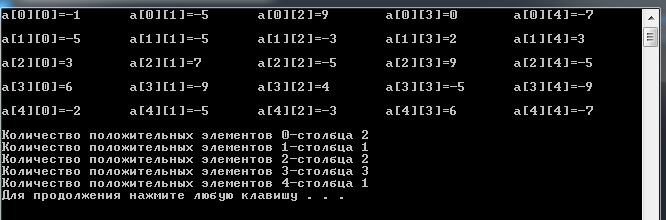
system("pause");

return 0;

}

* Результат вычислений





3.Составить программу для вычисления средних арифметических значений положительных элементов каждого столбца двумерного массива, содержащего 6 столбцов и три строки. При условии, что в каждом столбце есть хотя бы один положительный элемент.

* Код программы

#include <iostream>

#include<ctime>

using namespace std;

int main() {

setlocale(LC\_ALL, "RUS");

srand(time(NULL));

const int N = 3, M = 4;

int a[N][M] = {};

int kol;

float sum,avg;

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

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

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

cout << "a[" << i << "][" << j << "]=" << a[i][j] <<"\t";}

cout << endl;}

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

sum = 0;

kol = 0;

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

if (a[i][j] > 0) {

sum += a[i][j];

kol++;}

}

if (kol > 0){

avg = sum / kol;

cout << "B " << j << " столбце среднее арифметическое положительных элементов = " << avg << endl;}

else{

cout << "B " << j << " стобце нет положительных элемментов" << endl;

}

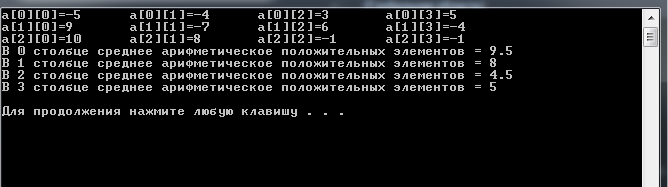
}

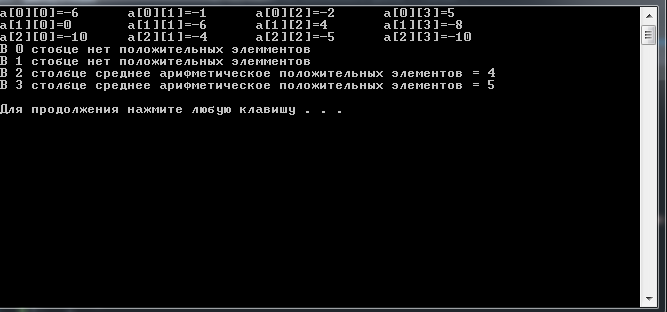
cout << endl;

system("pause");

return 0;}

* Результат вычислений





4.Дана действительная квадратная матрица. Заменить нулями все элементы, расположенные на главной диагонали и выше нее.

* Код программы

#include<iostream>

#include<ctime>

using namespace std;

int main() {

setlocale(LC\_ALL, "Rus");

srand(time(NULL));

const int N = 5;

int a[N][N] = {};

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

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

a[i][j] = rand() % 11;

cout << "a[" << i << "][" << j << "]=" << a[i][j] << "\t";}

cout << endl;}

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

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

if (i <= j) {

a[i][j] = 0;}}}

cout << endl;

cout << "После перестановки масив выглядит так:" << "\n\n";

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

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

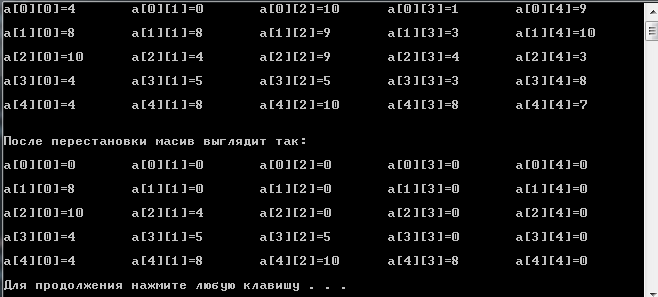
cout << "a[" << i << "][" << j << "]=" << a[i][j] << "\t";}

cout << endl;}

system("pause");

return 0;}

* Результат вычислений



5.Даны 8 действительных чисел х1, х2, …, х8. Получить квадратную матрицу 8х8

х1 х2 … х8

х12 х22 … х82

…..

х18 х28 … х88

* Код программы

#include <iostream>

#include <ctime>

using namespace std;

int main() {

setlocale(LC\_ALL, "RUS");

srand(time(NULL));

const int N = 5, M = 1;

double a[N][N] = {};

int b[N] = {};

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

b[i] = rand() % 11;

cout << "a[" << i << "]=" << b[i] << "\t";

}

cout << endl;

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

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

a[i][j] = pow(b[j],i+1);}

}

cout << "После перестановки массив выглядит так:" << endl;

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

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

cout << "a[" << i << "][" << j << "]=" << a[i][j] << "\t";

}

cout << "\n\n";

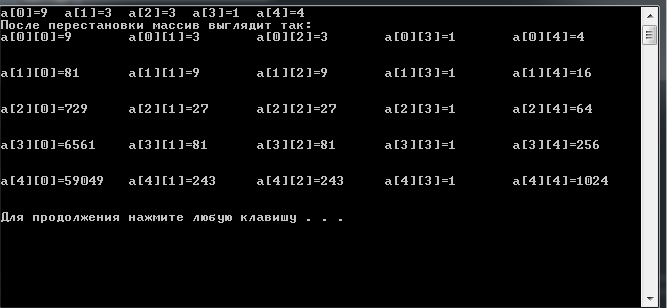
}

system("pause");

return 0;

}

* Результат вычислений



7.Дан двумерный массив, содержащий 3 строки и 4 столбца. Упорядочить массив по убыванию элементов 3-ей строки.

* Код программы

#include <iostream>

#include<ctime>

using namespace std;

int main() {

setlocale(LC\_ALL, "rus");

const int N = 3, M = 4;

int a[N][M], temp;

srand(time(NULL));

cout << "Массив размером 3х4: " << endl;

cout << endl;

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

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

a[i][j] = rand() % 11;

cout << "a[" << i << "][" << j << "]=" << a[i][j] << "\t";}

cout << endl;}

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

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

if (a[2][j] < a[2][j + 1]) {

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

temp = a[s][j];

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

a[s][j + 1] = temp;}}}}

cout << "\nРезультат:" << endl;

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

cout << endl;

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

cout << "a[" << i << "][" << j << "]=" << a[i][j] << "\t";

}

}

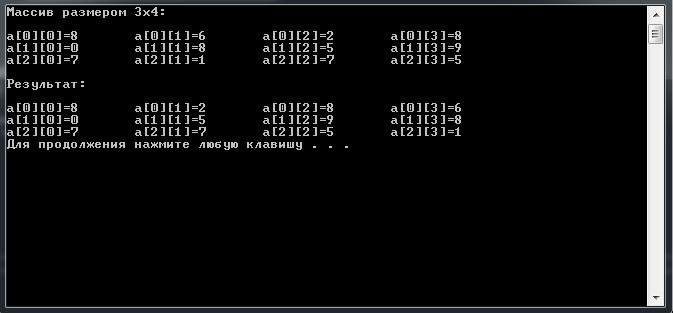
cout << endl;

system("pause");

return 0;

}

* Результат вычислений



8.Дан двумерный массив, содержащий 5 строк и 2 столбца. Упорядочить массив по возрастанию элементов 2-го столбца.

* Код программы

#include <iostream>

#include<ctime>

using namespace std;

int main() {

setlocale(LC\_ALL, "rus");

const int N = 5, M = 2;

int a[N][M], temp;

srand(time(NULL));

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

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

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

cout << "a[" << i << "][" << j << "]=" << a[i][j] << "\t";

}

cout << endl;}

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

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

if (a[i][1] > a[i + 1][1]) {

for (int s = 0; s < M; s++) {

temp = a[i][s];

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

a[i + 1][s] = temp;}}

}

} cout << endl;

cout << endl;

cout << "Результат:";

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

cout << endl;

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

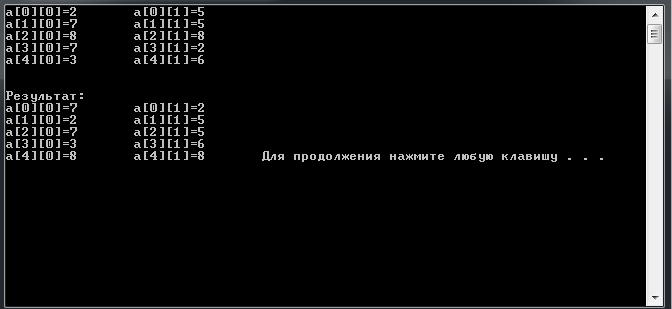
cout << "a[" << i << "][" << j << "]=" << a[i][j] << "\t";}}

system("pause");

return 0;

}

* Результат вычислений



1. Даны целые числа . Получить целочисленную матрицу b каждый элемент, которой определяется по следующей формуле , i,j=0,1,2.

* Код программы

#include <iostream>

#include<ctime>

using namespace std;

int main() {

setlocale(LC\_ALL, "rus");

srand(time(NULL));

const int N = 3;

int a[N] = {}, b[N][N] = {};

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

a[i] = rand() % 11;

cout << "a[" << i << "]=" << a[i] << "\t";}

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

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

b[i][j] = a[i] - (3 \* a[j]);}}

cout << endl;

cout << "Матрица b:" << endl;

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

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

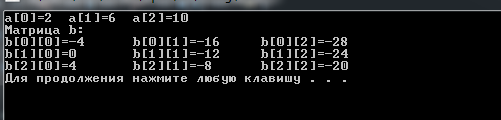
cout << "b[" << i << "][" << j << "]=" << b[i][j] << "\t";}

cout << endl;}

system("pause");

return 0;}

* Результат вычислений



1. Дана квадратная матрица А, содержащая 5 строк и 5 столбцов. Получить две квадратные матрицы В и С, элементы которых определяются по следующим формулам



* Код программы

#include<iostream>

#include<ctime>

using namespace std;

int main() {

setlocale(LC\_ALL, "rus");

srand(time(NULL));

const int N = 5;

int A[N][N]={}, B[N][N]={}, C[N][N]={};

cout << "Матрица A:" << endl;

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

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

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

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

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

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

if (j >= 0) {

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

if (j < i) {

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

cout << "Матрица B:" << endl;

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

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

cout << "B["<<i<<"]["<<j<<"]="<<B[i][j] << "\t";}

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

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

if (j < i) {

C[i][j] = A[i][j];}

if (j >= i) {

C[i][j] = -A[i][j];}}}

cout << "Матрица C:" << endl;

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

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

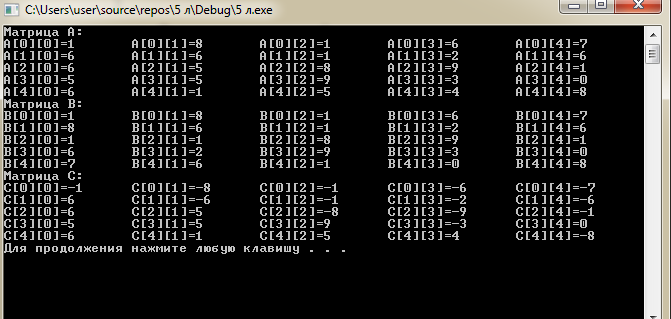
cout <<" C["<<i<<"]["<<j<<"]="<<C[i][j] << "\t";}}

system("pause");

return 0;

}

* Результат вычислений



1. Найти наибольший элемент главной диагонали матрицы С размером 4х4 и вывести на печать всю строку, в которой он находится.

* Код программы

#include<iostream>

#include<ctime>

using namespace std;

int main() {

setlocale(LC\_ALL, "rus");

srand(time(NULL));

const int N = 4;

int a[N][N] = {};

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

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

a[i][j] = rand() % 101;

cout << "a[" << i << "][" << j << "]=" << a[i][j] << "\t";}

cout << endl;}

int max = a[0][0];

int index = 0;

for (int i = 1; i < N; i++) {

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

max = a[i][i];

index = i;}}

cout << "\nНаибольший элемент равен:" << max << endl;

cout << "Наибольший элемент находится на строчке:" << endl;

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

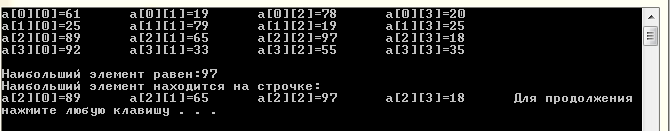
cout << "a[" << index << "][" << i << "]=" << a[index][i] << "\t";}

system("pause");

return 0;

}

* Результат вычислений



1. Перемножить матрицы А размером nхm и В размером mхl. Элементы результирующей матрицы получить с помощью следующей формулы 

* Код программы

#include<iostream>

#include<ctime>

using namespace std;

int main(){

setlocale(LC\_ALL, "rus");

srand(time(NULL));

const int N = 100,M=100,L=100;

int n, m, l;

int a[N][M] = {}, b[M][L] = {}, c[N][L] = {};

cout << "Введите размер первой матрицы:";

cin >> n >> m;

cout << "Введите размер второй матрицы(m из матрицы а равна m из матрицы b):";

cin >> l;

cout << "Матрица а:" << endl;

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

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

a[i][j] = rand() % 11;

cout << "a[" << i << "][" << j << "]=" << a[i][j] << "\t";

}

cout << endl;

}

cout << endl;

cout << "Матрица b:" << endl;

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

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

b[i][j] = rand() % 11;

cout << "b[" << i << "][" << j << "]=" << b[i][j] << "\t";

}

cout << endl;

}

cout << endl;

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

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

c[i][j] = 0;

for (int k = 0; k < l ; k++) {

c[i][j] += a[i][k] \* b[k][j];

}

}

}

cout << endl;

cout << "Матрица с:" << endl;

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

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

cout << "c["<<i<<"]["<<j<<"]=" << c[i][j] << "\t";

}

cout << endl;

}

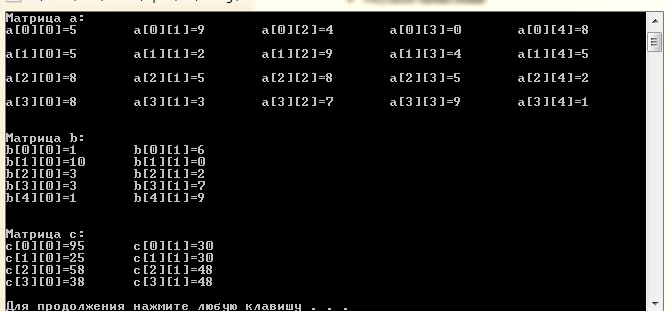
cout << endl;

system("pause");

return 0;

}

* Результат вычислений



1. Вычислить суммы элементов каждой строки матрицы А размером 6х6, определить наибольшее значение этих сумм.

* Код программы

#include<iostream>

#include<ctime>

using namespace std;

int main() {

setlocale(LC\_ALL, "rus");

srand(time(NULL));

const int N = 5;

int sum, max;

int a[N][N] = {};

int b[N] = {};

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

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

a[i][j] = rand() % 5;

cout << "a[" << i << "][" << j << "]=" << a[i][j] << "\t";}

cout << endl;

}

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

sum = 0;

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

b[i]+= a[i][j];}

cout << "Сумма элементов " << i << " строки равна: " << b[i] <<"\n\n";

sum = b[i];}

max = b[0];

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

if (b[i] > max)

max = b[i];}

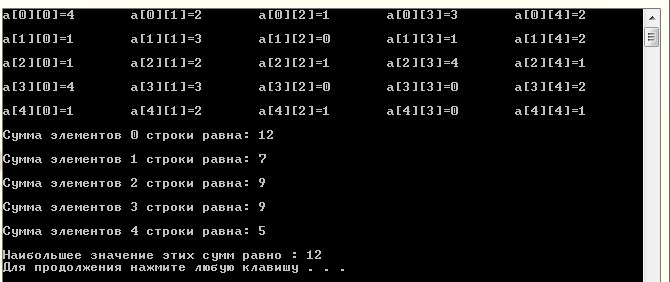
cout << "Наибольшее значение этих сумм равно : " << max << endl;

system("pause");

return 0;

}

* Результат вычислений



1. Дана действительная матрица размера 6х9. Найти среднее арифметическое наибольшего и наименьшего значений ее элементов.

* Код программы

#include<iostream>

#include<ctime>

using namespace std;

int main() {

setlocale(LC\_ALL, "rus");

const int N = 6, M = 5;

int a[N][M], max, min;

double sr;

srand(time(NULL));

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

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

a[i][j] = rand() % 101;

cout << "a[" <<i<<"]["<<j<<"]="<< a[i][j] << "\t";

}

cout << endl;

}

cout << endl;

max = a[0][0];

min = a[0][0];

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

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

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

max= a[i][j];

if (a[i][j] < min)

min = a[i][j];

}

}

sr = (max+min) / 2.;

cout << "Наибольший элемент равен:" << max << ",a наименьший элемент равен:" <<min << endl;

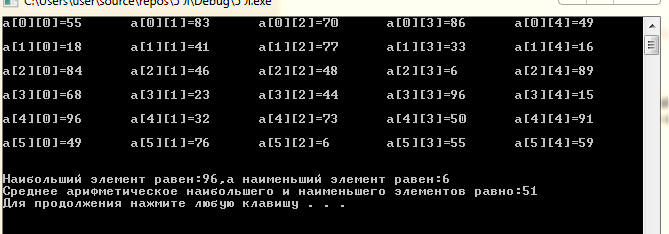
cout << "Cреднее арифметическое наибольшего и наименьшего элементов равно:" << sr << endl;

system("pause");

return 0;

}

* Результат вычислений



1. В квадратной матрице размера mxn найти значение наибольшего по модулю элемента матрицы, а также определить индексы этого элемента. Предполагается, что такой элемент единственный.

* Код программы

#include<iostream>

#include<ctime>

using namespace std;

int main() {

setlocale(LC\_ALL, "rus");

const int N = 100, M = 100;

int A[N][M], max, n, m, index1 = 0, index2= 0;

srand(time(NULL));

cout << "Введите размер матрицы: ";

cin >> n >> m;

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

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

A[i][j] = rand() % 301-200;

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

}

cout << endl;

}

cout << endl;

max = abs(A[0][0]);

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

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

if (abs(A[i][j]) > max) {

max = abs(A[i][j]);

index1 = i;

index2= j;

}

}

}

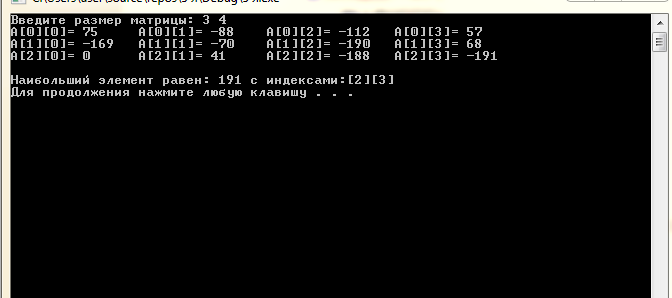
cout << "Наибольший элемент равен: "<< max <<" с индексами:["<<index1<<"]["<<index2<<"]"<<endl;

system("pause");

return 0;

}

* Результат вычислений



1. В данной действительной квадратной матрице порядка N найти сумму элементов строки, в которой расположен элемент с наименьшим значением. Предполагается, что такой элемент единственный.

* Код программы

#include<iostream>

#include<ctime>

using namespace std;

int main() {

setlocale(LC\_ALL, "rus");

const int N = 50, M = 50;

double A[N][M], min = 0, s = 0;

int n, x = 0, y = 0;

srand(time(NULL));

cout << "Введите число n " << endl;

cin >> n;

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

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

A[i][j] = rand() % 15;

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

}

cout << endl;

}

cout << endl;

min = A[0][0];

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

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

if (A[i][j] <= min) {

min = A[i][j];

x = i;

y = j;

}

}

}

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

s = s + A[x][i];

}

cout << "Cтрока №" << x << endl;

cout << "Сумма " << s << endl;

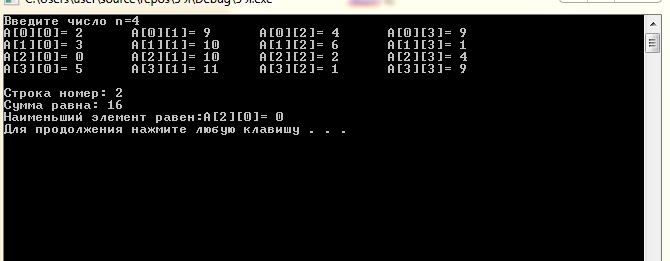
cout << "Наименьший элемент -> " << "A[" << x << "]" << "[" << y << "]= " << A[x][y] << endl;

system("pause");

return 0;

}

* Результат вычислений



1. Получить целочисленную квадратную матрицу порядка n, элементами которой являются числа 1, 2, 3, …, n2 , расположенные по спирали
2. 2 3 4 5

* Код программы

#include <iostream>

#include <cmath>

#include <ctime>

using namespace std;

int main() {

setlocale(LC\_ALL, "rus");

int n;

cout << "Введите размер квадратной матрицы: ";

cin >> n;

int Arr[50][50], d = n, k, count = 1;

for (int p = 0; count <= pow(n, 2); p++) {

for (int k = p; k < d; k++)

Arr[p][k] = count++;

for (int k = p + 1; k < d - 1; k++)

Arr[k][n - (p + 1)] = count++;

for (int k = n - (p + 1); k >= p; k--)

Arr[n - (p + 1)][k] = count++;

for (int k = n - (p + 2); k > p; k--)

Arr[k][p] = count++;

d--;

}

if (n % 2 == 1)

Arr[n / 2][n / 2] = pow(n, 2);

cout << endl;

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

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

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

}

cout << "\n\n\n";

}

system("pause");

return 0;

}

* Результат вычислений

