Министерство образования Республики Беларусь

г. Минск

Государственное учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

**Лабораторная работа №4**

**“** **Многомерные массивы.”**

**Учебная группа 230501**

Выполнил: Кочеров Роман Сергеевич

Проверил: Селезнев Александр Игоревич

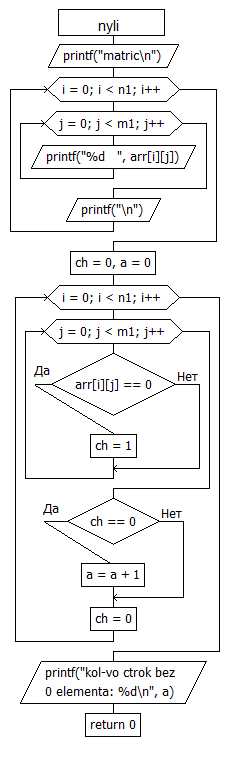
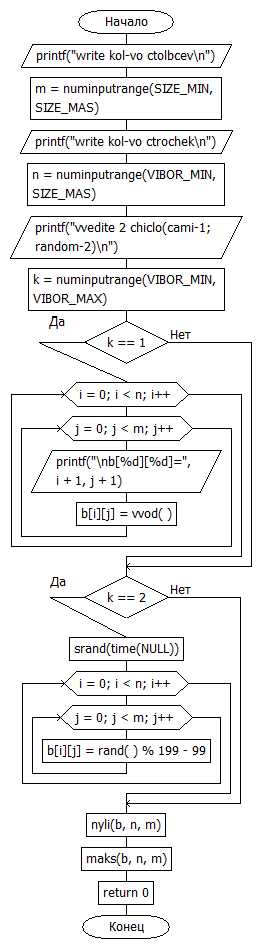
2022 год

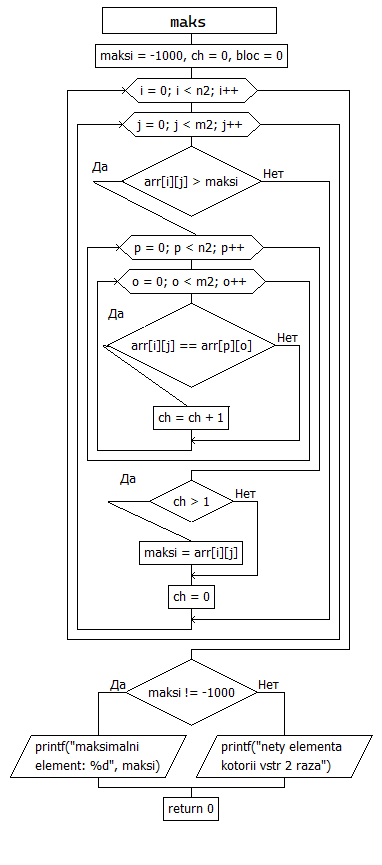
**Вариант 4**

**Задача 1**

Дана целочисленная прямоугольная матрица. Определить количество строк, не содержащих ни одного нулевого элемента; максимальное из чисел, встречающихся в заданной матрице более одного раза.

Ввод: n, m, k, \*b[m\*n](числа).





#include <stdio.h>

#define SIZE\_MAS 100

#define SIZE\_MIN 0

#define VIBOR\_MIN 1

#define VIBOR\_MAX 2

int numinputrange(int min, int max);

void nyli(int arr[SIZE\_MAS][SIZE\_MAS], int n1, int m1);

void maks(int arr[SIZE\_MAS][SIZE\_MAS], int n2, int m2);

int vvod();

int main() {

int b[SIZE\_MAS][SIZE\_MAS];

int n, m, k;

printf("write kol-vo ctolbcev\n");

m = numinputrange(SIZE\_MIN, SIZE\_MAS);

printf("write kol-vo ctrochek\n");

n = numinputrange(VIBOR\_MIN, SIZE\_MAS);

printf("vvedite 2 chiclo(cami-1; random-2)\n");

k = numinputrange(VIBOR\_MIN, VIBOR\_MAX);//тут выбор сам или нет

if (k == 1){

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

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

printf("\nb[%d][%d]=", i + 1, j + 1);//+1 чтобы нормально выводилось

b[i][j] = vvod();} }

if (k == 2){

srand(time(NULL));

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

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

b[i][j] = rand() % 199 - 99;}

nyli(b,n,m);

maks(b,n,m);

return 0;}

int numinputrange(int min, int max)

{

int var;

while ((scanf\_s("%d", &var) == 0) || (var < min) || (var > max) || getchar() != '\n')

{

printf("wrong input! try again\n");

rewind(stdin);

}

return var;

}

void nyli(int arr[SIZE\_MAS][SIZE\_MAS], int n1, int m1)

{

printf("matric\n");//просто вывод матрицы

for (int i = 0; i < n1; i++)

{

for (int j = 0; j < m1; j++)

printf("%3d ", arr[i][j]);

printf("\n");

}

int ch = 0, a = 0;

for (int i = 0; i < n1; i++)

{

for (int j = 0; j < m1; j++)

{

if (arr[i][j] == 0)//прокручивает строку, если нашёл 0 то меняет счётчик

ch = 1;

}

if (ch == 0)//если не нашёл ноли то счётскик = 0

a = a + 1;//кол-во строк без нолей

ch = 0;

}

printf("kol-vo ctrok bez 0 elementa: %d\n", a);

return 0;

}

void maks(int arr[SIZE\_MAS][SIZE\_MAS], int n2, int m2)

{

int maksi = -1000, ch = 0, bloc = 0;

for (int i = 0; i < n2; i++)//прокручивает массив

{

for (int j = 0; j < m2; j++)

{

if (arr[i][j] > maksi)

{

for (int p = 0; p < n2; p++)//ещё один массив для повторящек

{

for (int o = 0; o < m2; o++)

{

if (arr[i][j] == arr[p][o])

{

ch = ch + 1;

}

}

}

if (ch > 1)

{

maksi = arr[i][j];//если нашёл повторяшку, то перезапишет

}

ch = 0;

}

}

}

if (maksi != -1000)//такое макси если не нашёл повторяшек

printf("maksimalni element: %d", maksi);

else

printf("nety elementa kotorii vstr 2 raza");

return 0;

}

int vvod()

{

int var;

while ((scanf\_s("%d", &var) == 0) || getchar() != '\n')

{

printf("wrong input! try again\n");

rewind(stdin);

}

return var;

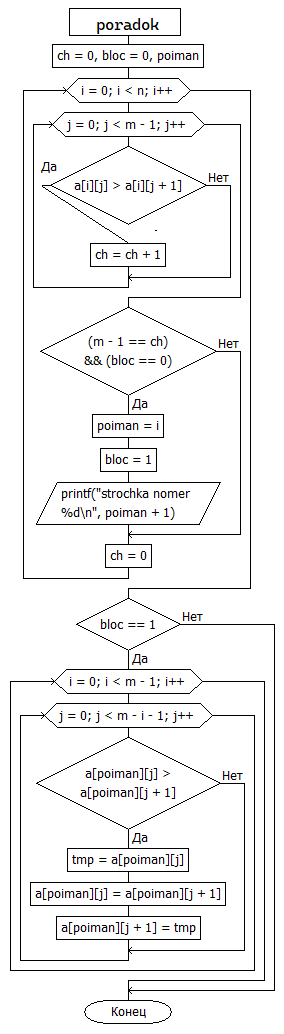
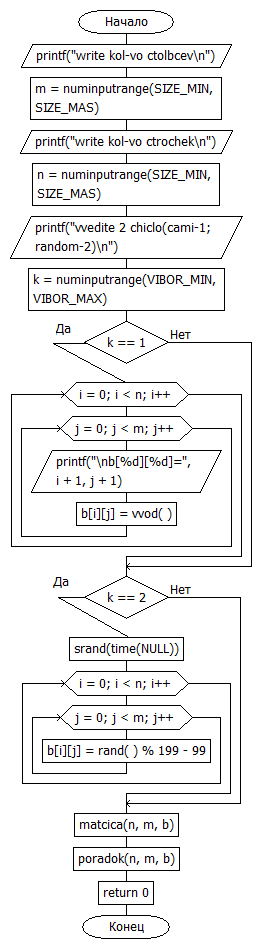
}



**Задача 2**

Найти в матрице первую строку, все элементы которой упорядочены по убыванию. Изменить упорядоченность элементов этой строки на обратную.

Ввод: n, m, k, b[m\*n](числа).



#include <stdio.h>

#define SIZE\_MAS 100

#define SIZE\_MIN 0

#define VIBOR\_MIN 1

#define VIBOR\_MAX 2

int numinputrange(int min, int max);

void matcica(int n1, int m1, int arr[SIZE\_MAS][SIZE\_MAS]);

void poradok(int n, int m, int a[SIZE\_MAS][SIZE\_MAS]);

int vvod();

int main() {//всё как в аервой проге

int b[SIZE\_MAS][SIZE\_MAS];

int n, m, k;

printf("write kol-vo ctolbcev\n");

m = numinputrange(SIZE\_MIN, SIZE\_MAS);

printf("write kol-vo ctrochek\n");

n = numinputrange(SIZE\_MIN, SIZE\_MAS);

printf("vvedite 2 chiclo(cami-1; random-2)\n");

k = numinputrange(VIBOR\_MIN, VIBOR\_MAX);//тут выбор сам или нет

if (k == 1) {

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

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

printf("\nb[%d][%d]=", i + 1, j + 1);

b[i][j] = vvod(); } }

if (k == 2){

srand(time(NULL));

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

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

b[i][j] = rand() % 199 - 99; }

matcica(n, m, b);

poradok(n, m, b);

return 0;}

int numinputrange(int min, int max)

{

int var;

while ((scanf\_s("%d", &var) == 0) || (var < min) || (var > max) || getchar() != '\n')

{

printf("wrong input! try again\n");

rewind(stdin);

}

return var;

}

void matcica(int n1, int m1, int arr[SIZE\_MAS][SIZE\_MAS]) {

printf("matric\n");

for (int i = 0; i < n1; i++)

{

for (int j = 0; j < m1; j++)

printf("%3d ", arr[i][j]);

printf("\n");

}

}

void poradok(int n, int m, int a[SIZE\_MAS][SIZE\_MAS])

{

int ch = 0, bloc = 0, poiman;

for (int i = 0; i < n; i++)//поиск первой строки в которой эл-ты по убыванию

{

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

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

{

ch = ch + 1;

}

if ((m - 1 == ch) && (bloc == 0))

{

poiman = i;

bloc = 1;

printf("strochka nomer %d\n", poiman+1);

}

ch = 0;

}

if (bloc == 1)

{

for (int i = 0; i < m - 1; i++)//перестановка пузырьком(максимальный в конец)

{

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

if (a[poiman][j] > a[poiman][j + 1])

{

int tmp = a[poiman][j];

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

a[poiman][j + 1] = tmp;

}

}

}

}

printf("matric 2\n");//вывод матрицы

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

{

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

printf("%3d ", a[i][j]);

printf("\n");

}

}

int vvod()

{

int var;

while ((scanf\_s("%d", &var) == 0) || getchar() != '\n')

{

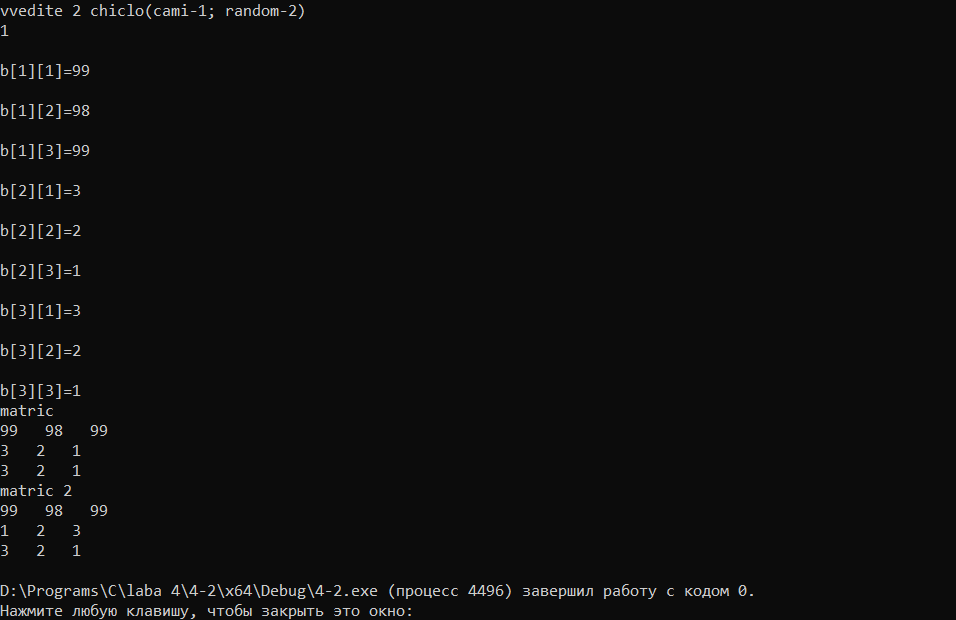
printf("wrong input! try again\n");

rewind(stdin);

}

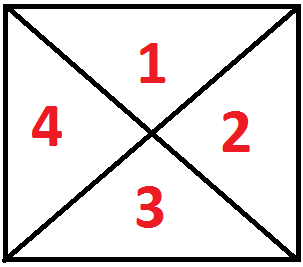
return var;

}

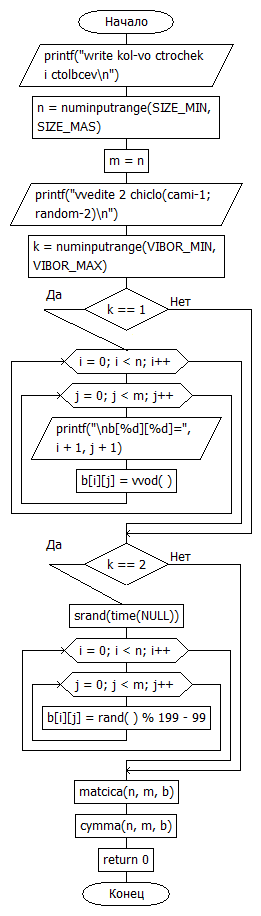


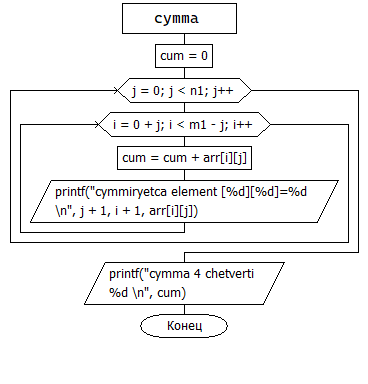
**Задача 3**

В квадратной матрице размером NxN найти сумму элементов в 4-ой области.



Ввод: n, k, b[n\*n] (числа).





#include <stdio.h>

int numinputrange(int min, int max)

{

int var;

while ((scanf\_s("%d", &var) == 0) || (var < min) || (var > max) || getchar() != '\n')

{

printf("wrong input! try again\n");

rewind(stdin);

}

return var;

}

void matcica(int n1, int m1, int arr[100][100]) {

printf("matric\n");

for (int i = 0; i < n1; i++)

{

for (int j = 0; j < m1; j++)

printf("%d ", arr[i][j]);

printf("\n");

}

}

void cymma(int n1, int m1, int arr[100][100])

{

int cum = 0;

for (int j = 0; j < n1; j++)

{

for (int i = 0+j; i < m1-j; i++)//откидываем столбы сверху и снизу на j, на середине i станет слишком большим и не будет запускать

{

cum = cum + arr[i][j];

printf("cymmiryetca element [%d][%d]=%d \n", i, j, arr[i][j]);

}

}

printf("cymma 4 chetverti %d \n", cum);

}

int main() {//всё как во 2 задаче но теперь тока n

int b[100][100];

int n, m, k;

printf("write kol-vo ctrochek i ctrochec\n");

n = numinputrange(0, 100);

m = n;

printf("vvedite 2 chiclo(cami-1; random-2)\n");

k = numinputrange(1, 2);//тут выбор сам или нет

if (k == 1)

{

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

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

{ printf("\nb[%d][%d]=", i + 1, j + 1);

b[i][j] = numinputrange(-10000000, 100000000);}

}

if (k == 2)

{ srand(time(NULL));

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

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

b[i][j] = rand() % 199 - 99;}

matcica(n, m, b);

cymma(n, m, b);

return 0;}

