

Многомерные массивы.

Цель работы: освоить работу с многомерными массивами, решение задач по вариантам.

Краткие теоретические сведения

Массивы в Си могут иметь несколько индексов. Многомерные массивы часто применяются для представления таблиц, состоящих из значений, упорядоченных по *строкам* и *столбцам*. Многомерные массивы – это *массив массивов*, т.е. массив элементами которого являются массивы.

Многомерные массивы объявляются точно так же, как и одномерные, только после имени массива ставится *более одной пары* квадратных скобок.

Пример объявления многомерных массивов:

```
double array[30][10]; // матрица с 30 строками и 10 столбцами
int buffer[1][7]; // матрица с 1 строкой и 7 столбцами
int bad1[0][5]; // ошибка: размерность не может быть нулевой
int bad2[0][3][0]; // ошибка: размерность не может быть нулевой
```

Ввод с клавиатуры, вывод на экран и другие действия с матрицей, состоящей из n строк и m столбцов. Особых отличий работы с многомерными массивами от одномерных нет. При обращении к элементам многомерного массива необходимо указать все его индексы.

Пример ввода с клавиатуры элементов матрицы:

```
int i,j;
float b[100][100];
for (i=0;i<n;i++)
    for (j=0;j<m;j++)
    {
        printf("Vvedite element [%d,%d]\n", i+1, j+1);
        scanf("%f",&b[i][j]);
    }
```

Пример вывода на экран элементов матрицы в табличной форме:

```
int i,j;
float b[100][100];
printf("MATRICA: \n");
for (i=0;i<n;i++)
{
    for (j=0;j<m;j++)
        printf("%.2f", b[i][j]);
    printf("\n");
}
```

Пример нахождения суммы элементов матрицы:

```
int i,j;
float b[100][100];
float s=0;
for (i=0;i<n;i++)
    for (j=0;j<m;j++)
        s+=b[i][j]
printf("Summa elementov matrici = %.2f\n", s);
```

Пример нахождения максимального элемента матрицы:

```
int i,j;
float b[100][100];
float max=b[0][0];
for (i=0;i<n;i++)
    for (j=0;j<m;j++)
        if (b[i][j]>max)
            max=b[i][j];
printf("Max element matrici = %.2f\n", max);
```

Пример нахождения минимального элемента матрицы:

```
int i,j;
float b[100][100];
float min=b[0][0];
for (i=0;i<n;i++)
    for (j=0;j<m;j++)
        if (b[i][j]<min)
            min=b[i][j];
printf("Min element matrici = %.2f\n", min);
```

Пример нахождения суммы каждой строки матрицы

```
int i,j;
float s;
float b[100][100];
for (i=0;i<n;i++)
{
    s=0;
    for (j=0;j<m;j++)
        s+=b[i][j];
    printf("Summa elementov %d-stroki matrici = %.2f\n", i+1, s);
}
```

Пример нахождения произведения каждого столбца матрицы

```
int i,j;
float p;
float b[100][100];
for (j=0;j<m;j++)
{
    p=1;
```

```

    for (i=0;i<n;i++)
        p*=b[i][j];
    printf("Proizvedenie elementov %d-stolbca matrici = %.2f\n", j+1, p);
}

```

Пример подсчета количества положительных элементов на главной диагонали

```

int i, j, kol=0;
float b[100][100];
for (i=0;i<n;i++)
{
    if (b[i][i]>0)
        kol++;
}
printf("Kol-vo pologitelnih elementov na glavnoi diagonal = %.d\n", kol);

```

Пример 1: Дана матрица вещественных чисел $N \times N$. Количество строк и столбцов N задается пользователем. Подсчитать сумму элементов матрицы, найти минимальный и максимальный элементы матрицы, найти сумму каждой строки матрицы, произведение каждого столбца матрицы и количество положительных элементов на главной диагонали.

```

#include <stdio.h>
#include <conio.h>
void main ()
{
    int i,j, kol=0, n;
    float b[30][30], s=0, p, max, min;
    do{
        printf("Vvedite kol-vo strok i stolbcov matrici (<30)\n");
        scanf("%d",&n);
    } while (n>=30);

    for (i=0;kn;i++)
        for (j=0;j<n;j++)
        {
            printf("Vvedite element [%d,%d]\n", i+1, j+1);
            scanf("%f",&b[i][j]);
        }

    for (i=0;i<n;i++)
        for (j=0;j<m;j++)
            s+=b[i][j]
    printf("Summa elementov matrici = %.2f\n", s);

    max=b[0][0];
    for (i=0;i<n;i++)
        for (j=0;j<m;j++)

```

```

        if (b[i][j]>max)
            max=b[i][j];
printf("Max element matrici = %.2f\n", max);

min=b[0][0];
for (i=0;i<n;i++)
    for (j=0;j<m;j++)
        if (b[i][j]<min)
            min=b[i][j];
printf("Min element matrici = %.2f\n", min);

for (i=0;i<n;i++)
{
    s=0;
    for (j=0;j<m;j++)
        s+=b[i][j];
    printf("Summa elementov %d-stroki matrici = %.2f\n", i+1, s);
}

for (j=0;j<m;j++)
{
    p=1;
    for (i=0;i<n;i++)
        p*=b[i][j];
    printf("Proizvedenie elementov %d-stolbca matrici = %.2f\n", j+1, p);
}

for (i=0;i<n;i++)
{
    if (b[i][i]>0)
        kol++;
}
printf("Kol-vo pologitelnih elementov na glavnoi diagonal = %.d\n", kol);

printf("MATRICA: \n");
for (i=0;i<n;i++)
{
    for (j=0;j<m;j++)
        printf("%.2f", b[i][j]);
    printf("\n");
}
}

```

Практическая часть

Упражнение 1

1. Дана матрица вещественных чисел $N \times M$. Количество строк N и столбцов M задается пользователем. Найти максимальный и минимальный элемент в каждой строке и поменять их местами. Найти сумму элементов на главной диагонали.

2. Характеристикой строки целочисленной матрицы назовем сумму ее отрицательных четных элементов. Переставляя строки заданной матрицы, расположить их в соответствии с убыванием характеристик.

3. Упорядочить строки целочисленной прямоугольной матрицы по возрастанию количества одинаковых элементов в каждой строке. Найти номер первого из столбцов, не содержащих ни одного отрицательного элемента.

4. Дана целочисленная прямоугольная матрица. Определить количество строк, содержащих хотя бы один нулевой элемент; номер столбца, в которой находится самая длинная серия одинаковых элементов.

5. Дана целочисленная квадратная матрица. Определить: сумму элементов в тех строках, которые не содержат отрицательных элементов; минимум среди сумм элементов диагоналей, параллельных главной диагонали матрицы.

6. Дана целочисленная квадратная матрица. Определить: сумму элементов в тех столбцах, которые не содержат отрицательных элементов; минимум среди сумм модулей элементов диагоналей, параллельных побочной диагонали матрицы.

7. Проверить, все ли строки матрицы содержат хотя бы один отрицательный элемент. Если да, то изменить знаки всех элементов матрицы на обратные.

8. Проверить, все ли строки матрицы содержат хотя бы один нулевой элемент. Если нет, то заменить значения всех отрицательных элементов матрицы на нулевые.

9. Найти в матрице первый столбец, все элементы которого положительны. Знаки элементов предыдущего столбца изменить на противоположные.

10. Найти в матрице первый столбец, все элементы которого отрицательны, и среднее арифметическое этих элементов. Вычесть полученное значение из всех элементов матрицы.

11. Найти в матрице первый столбец, все элементы которого равны нулю. Знаки элементов строки с таким же номером изменить на противоположные.

Упражнение 2

1. Двумерный массив, содержащий равное число строк и столбцов, называется магическим квадратом, если суммы чисел, записанных в каждой строке, каждом столбце и каждой из двух больших диагоналей, равны одному и тому же числу. Определить, является ли данный массив A из n строк и n столбцов магическим квадратом.

2. Элемент матрицы назовем седловой точкой, если он наименьший в своей строке и наибольший (одновременно) в своем столбце (или наоборот, наибольший в своей строке и наименьший в своем столбце). Для заданной целой матрицы размером 10×12 напечатать индексы всех ее седловых точек.

3. Дана матрица размером 6×6 . Найти сумму наименьших элементов ее нечетных строк и наибольших элементов ее четных строк.

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

5. Дана целочисленная прямоугольная матрица. Определить количество столбцов, не содержащих ни одного нулевого элемента.

6. Характеристикой строки целочисленной матрицы назовем сумму ее положительных четных элементов. Переставляя строки заданной матрицы, расположить их в соответствии с ростом характеристик.

7. Дана целочисленная прямоугольная матрица. Определить количество столбцов, содержащих хотя бы один нулевой элемент; номер строки, в которой находится самая длинная серия одинаковых элементов.

8. Дана целочисленная квадратная матрица. Определить произведение элементов в тех строках, которые не содержат отрицательных элементов; максимум среди сумм элементов диагоналей, параллельных главной диагонали матрицы.

9. Для заданной матрицы размером 8×8 найти такие k , что k -я строка матрицы совпадает с k -м столбцом. Найти сумму элементов в тех строках, которые содержат хотя бы один отрицательный элемент.

10. Характеристикой столбца целочисленной матрицы назовем сумму модулей его отрицательных нечетных элементов. Переставляя столбцы заданной матрицы, расположить их в соответствии с ростом характеристик. Найти сумму элементов в тех столбцах, которые содержат хотя бы один отрицательный элемент.

Упражнение 3

1. Найти в матрице первую строку, все элементы которой положительны, и сумму этих элементов. Уменьшить все элементы матрицы на эту сумму.
2. Найти в матрице первую строку, все элементы которой отрицательны. Увеличить все элементы матрицы на значение первого элемента найденной строки.
3. Найти в матрице первую строку, все элементы которой упорядочены по возрастанию. Изменить упорядоченность элементов этой строки на обратную.
4. Найти в матрице первую строку, все элементы которой упорядочены по убыванию. Изменить упорядоченность элементов этой строки на обратную.
5. Проверить, есть ли в матрице хотя бы одна строка, содержащая положительный элемент, и найти ее номер. Знаки элементов предыдущей строки изменить на противоположные.
6. Проверить, есть ли в матрице хотя бы одна строка, содержащая отрицательный элемент, и найти ее номер. Все элементы столбца с таким же номером уменьшить вдвое.
7. Проверить, есть ли в матрице хотя бы одна строка, содержащая элемент, равный нулю, и найти ее номер. Уменьшить все элементы матрицы на значение первого элемента найденной строки.
8. Найти в матрице первую строку, все элементы которой равны нулю. Все элементы столбца с таким же номером уменьшить вдвое.
9. Проверить, все ли строки матрицы упорядочены по убыванию. Если нет, найти первую неупорядоченную строку и упорядочить.
10. Проверить, все ли строки матрицы содержат хотя бы один положительный элемент. Если да, то изменить знаки всех элементов матрицы на обратные.