

Упражнение 6 – Многомерни масиви

Многомерен масив

Многомерният масив предоставя възможност за съхранение на данни, зависещи от повече от една променлива. Пример за многомерни масиви са числовите матрици от математика, както и представянето на изображенията в компютърните системи. Многомерните масиви подобно на едномерните в програмния език C съдържат само еднотипови данни, като достъп до полетата на този масив се осъществява чрез индекси за всяко едно измерение.

Дефиниране на масиви в C

В програмния език C многомерните масиви се използват чрез:

```
<тип>  
<име_на_масив>[<размер_на_измерение_1>][<размер_на_измерение_2>]...[<размер_на_измерение_N>];
```

Тук **тип** не се различава по значение от типовете данни, които се използват за дефинирането на променливи. и могат да бъдат използвани всички познати типове от данни.

Масивът като всяка друга променлива изисква да има **име**. Това име е нужно, за да може да се осъществи достъп до елементите на масива.

По време на дефинирането на многомерен масив е необходимо да се зададе **размерите на всяко измерение**. Размерите, както при едномерните масиви, представляват положителни константни цели числа и размерът им не може да бъде променян по време на изпълнението на програмата.

Пример за дефиниране на масив:

```
float measurements[100][100];
```

Начални стойности на масив

Начални стойности могат да бъдат заделени чрез изреждане на началните стойности по следния начин (пример за двумерен масив:

```
int values[5][5]={1, 2, 3, 4, 5},{1, 2, 3, 4, 5},{1, 2, 3, 4, 5},{1, 2, 3, 4, 5},{1, 2, 3, 4, 5};
```

Когато броят на началните стойности е по-малък от броят на всички елементи, то останалите елементи приемат начална стойност 0;

Достъп до елементите на масив

Работата с елементите на многомерните масиви не се различава като начин от работата с едномерните масиви, но тук се изисква използването на повече индекси (съответстващи на броя размерности на масива). Индексите отново са цяли неотрицателни числа, което съответства на позицията на съответния елемент в съответното измерение в масива. Последният елемент в масива има индекс **размер-1 (като това е стойността на размера за всяко едно измерение)**.

Работа с многомерни масиви

За работата с многомерни масиви се използват цикли(съответстващи на броя на размерностите) за достъп до елементите на масива, където броячите са индексите на съответния елемент. В следващият пример е показано въвеждането на стойностите от масива от клавиатурата и извеждането им в обратен ред на екрана:

```
#include <stdio.h>

int main()
{
    int i,j;
    int array[10][10];
    for(i = 0; i < 10; i++)
    {
        for(j = 0; j < 10; j++)
        {
            printf("Въведете стойност за елемент %d,%d : ",i,j);
            scanf("%d",&array[i][j]);
        }
    }
    printf("Извеждане на елементите на масива в обратен ред:\n");
    for(i = 10-1; i>=0; i--)
    {
        for(j = 10-1; j>=0; j--)
            printf("%d ",array[i][j]);
        printf("\n");
    }
    return 0;
}
```

Задачи за изпълнение

1. Да се изведат стойностите от главния диагонал на квадратна матрица $N \times N$, като $N < 100$
2. Стойностите от второстепенния диагонал на квадратна матрица $N \times N$, като $N < 100$, да се умножат по 3. Преобразуваната матрица да се изведе на екрана.
3. Нека имаме 5 града: София, Пловдив, Бургас, Варна и Плевен. Между всеки два града имаме размерност в километри. Като разполагаме с тази информация да се изчисли дължината на произволно зададен маршрут до 8 дестинации на екрана (През даден град може да се минава няколко пъти). Попитайте ръководителя на упражнението как може да се представят дестинациите като двумерна матрица.