

# مبانی برنامه نویسی

مدرس: سعدون عزیزی

[s.azizi@uok.ac.ir](mailto:s.azizi@uok.ac.ir)

مرکز آموزش های الکترونیکی

تأبستان ۹۶

# سرفصل مطالب

- آشنایی با کامپیوتر و الگوریتم
- مقدمه‌ای بر برنامه‌نویسی C
- محاسبات
- ورودی/خروجی
- حلقه‌ها
- دستورات شرطی
- توابع
- آرایه‌ها
- کاراکترها و رشته‌ها
- اشاره‌گرها
- ساختار
- فایل‌ها

## آرایه ها

- به منظور نگهداری مجموعه ای از داده های هم نوع در حافظه از ساختار آرایه استفاده می کنیم
- به این ترتیب که مجموعه داده ها در خانه های حافظه پشت سر هم و تحت یک نام مشترک ذخیره می گردند و به وسیله اندیس از یکدیگر تمایز داده می شوند
- برای کار با آرایه، ابتدا باید آن را معرفی کنیم تا فضای لازم برای آن تخصیص داده شود. پس از معرفی، می توان به تک تک عناصر آن دسترسی پیدا کرد و مقادیر آنها را تعیین نموده و در صورت لزوم بازیابی کرد

# آرایه یک بعدی

□ معرفی آرایه یک بعدی

[اندازه آرایه] نام آرایه نوع عناصر آرایه

□ اندازه آرایه توسط یک عبارت صحیح مثبت اعلام می شود. اندازه آرایه نباید متغیر باشد. ولی می تواند شامل ثابت سمبولیک یا شناسه ثابت باشد.

□ مثال:

```
int a[10];  
float x[20] , y[50];  
char text[40];
```

<b>a</b>	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]
	?	?	?	?	?	?	?	?	?	?

## آرایه یک بعدی (دسترسی به عناصر)

□ دسترسی به عناصر یک آرایه یک بعدی

[اندیس] نام آرایه

□ اندیس باید از نوع صحیح باشد

□ اندیس می تواند ثابت، متغیر و یا عبارت محاسباتی باشد

□ مثال:

```
int a[10];
```

```
a[3]=20;
```

## آرایه یک بعدی (مثال)

```
#include<stdio.h>
main()
{
    int i,count=0;
    float g[10],sum=0.0, ave;
    for(i=0; i<10; i++) {
        printf("enter the grade of student <%d>:",i+1);
        scanf("%g",&g[i]);
        sum+=g[i];
    }
    ave=sum/10;
    for(i=0; i<10; i++)
        if(g[i]>ave)
            count++;
    printf("\nThe average is: %g",ave);
    printf("\nThe number of grades above the average is: %d",count);
}
```

## آرایه یک بعدی (مقداردهی اولیه)

مثال:

```
int month[12]={31,31,31,31,31,31,30,30,30,30,30,29};
```

□ اگر تعداد عناصر آرایه بیش از تعداد مقادیر باشد، مقدار عناصر باقیمانده برابر صفر خواهد بود

```
int month[12]={31,31,31,31,31,31,30,30,30,30};
```

```
month[10] → 0
```

□ به کمک دستور زیر می توان به تمام عناصر آرایه مقدار صفر نسبت داد:

```
int month[12]={0};
```

# آرایه چند بعدی

□ معرفی آرایه چند بعدی:

[اندازه بعد n ام] ... [اندازه بعد دوم] [اندازه بعد اول] نام آرایه نوع عناصر آرایه

□ مثال:

```
int x[10][20];
```

```
float y[3][4][5];
```

```
char book[20][50];
```



## آرایه چند بعدی (مثال)

```
#include<stdio.h>
#define N 3
main()
{
    int mat[N][N],i,j,odd=1;
    for(i=0; i<N; i++)
        for(j=0; j<N; j++) {
            mat[i][j]=odd;
            odd+=2;
        }
    for(i=0; i<N; i++) {
        for(j=0; j<N; j++)
            printf("%5d",mat[i][j]);
        printf("\n");
    }
}
```

خروجی:

1	3	5
7	9	11
13	15	17

## آرایه چند بعدی (مقداردهی اولیه)

روش اول:

```
int x[3][4] = {1,2,3,4,5,6,7,8,9,10,11,12};
```

روش دوم:

```
int x[3][4] = { {1,2,3,4} , {5,6,7,8} , {9,10,11,12} };
```

□ در مقداردهی به صورت گروه‌بندی، چنانچه مقادیر لیست شده در گروهی کمتر از تعداد عناصر آن گروه باشد، مقدار عناصر اضافی صفر می‌شود.

```
int x[3][4] = { {1,2,3} , {5,6,7,8} , {9,10,11} };
```

## ارسال آرایه به تابع

- از آنجا که نام آرایه به تنهایی (بدون اندیس)، به آدرس شروع آرایه اشاره می کند، کافی است که هنگام فراخوانی تابع، نام آرایه در محل آرگومان قرار گیرد.
- ارسال آرایه به تابع از نوع فراخوانی با ارجاع است

# ارسال آرایه به تابع

سورس کدها در محیط DeV-C++