

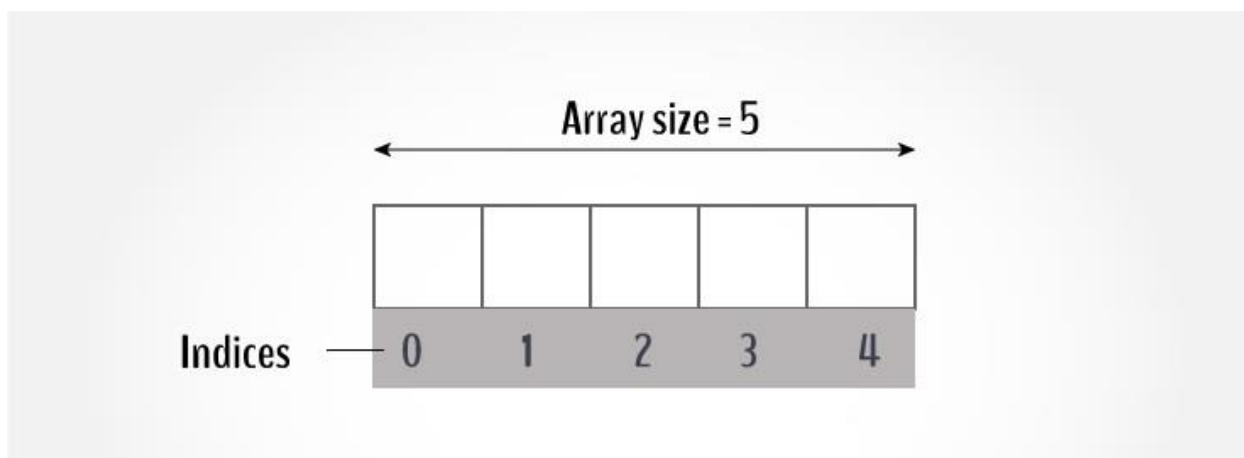
به نام خدا

دستور کارگاه کامپیوتر شماره ۶

آشنایی با آرایه ها در زبان C

در این جلسه شما با آرایه^۱ ها و نوع خاصی از آن ها یعنی رشته^۲ ها آشنا خواهید شد.

تعریف آرایه: تعدادی داده های پشت سر هم در حافظه که همگی از یک نوع^۳ هستند.



نکته: شماره اندیس آرایه به طول n از 0 شروع و تا $n-1$ ادامه دارد.

برای تعریف و مقدار دهی آرایه ها در زبان C روش های متعددی وجود دارد. در کد زیر تعدادی از این روش ها ذکر شده اند.

```
✓ int a[] = { 5, 3, 2 };  
✓ int b[3];  
✓ int c[3] = { 6, 5 };  
✓ b[0] = 2;
```

```
✓ int d[3];  
✗ d = { 4, 5, 6 };  
✓ int e[3];  
✗ e[3] = 4;
```

نکته: مقداردهی آرایه با استفاده از $\{\}$ تنها هنگام تعریف آرایه مجاز است.

فکر کنید! چرا مجاز به مقداردهی آرایه به صورت ذکر شده در خط آخر کد بالا نیستیم؟ کد بالا را اجرا کنید. چه اتفاقی می افتد؟

¹ array

² string

³ type

نکته: طول آرایه‌ها باید هنگام در زمان `compile` تعیین شود. بدین معنا که طول آرایه در زمان اجرای برنامه نمی توان تعریف یا تغییر داده شود. با توجه به این موضوع طول آرایه می‌توان به صورت متغیر `const` تعریف شود.

۱. انجام دهید!

- ۱) یک آرایه از نوع `int` به طول ۵ تعریف کنید. طول آرایه را با استفاده از متغیر `const` تعریف کنید.
- ۲) با استفاده از حلقه `for` و دستور `scanf`، ۵ عدد از کاربر دریافت نموده و در خانه‌های آرایه ذخیره کنید.
- ۳) با استفاده از دستور `printf` مقادیر وارد شده را به ترتیب معکوس در خروجی چاپ کنید.

راهنمایی :

```
scanf("%d", &a[i]);
```



نتیجه را به دستیاران آموزشی نشان دهید.

نکته: یک آرایه را نمی توان برابر یک آرایه ی دیگر گذاشت. مثلاً برنامه ی زیر خطای کامپایلی دارد.

```
int arr[4] = {1, 2, 3, 4};  
int arr_copy[4];  
arr_copy = arr;
```

سوال: پس چگونه می توانیم یک نسخه جدید از `arr` داشته باشیم؟

۲. انجام دهید! (تحلیل اطلاعات)

- ۱) از کاربردهای آرایه می‌توان به نگهداری اطلاعات جهت انجام تحلیل‌هایی بر روی آن‌ها ذکر کرد. به همین منظور یک آرایه به طول ۱۰ ایجاد کنید.
- ۲) با استفاده از حلقه `for` و دستور `scanf`، ۱۰ عدد از کاربر دریافت نموده و در خانه‌های آرایه ذخیره کنید.
- ۳) مقدار میانگین اعداد آرایه را محاسبه و چاپ نمایید. (برای این کار با استفاده از حلقه `for` جمع اعداد آرایه را حساب کرده و بر ۱۰ تقسیم کنید. برای دقیق‌تر بودن پاسخ از متغیر `float` برای نگه داری جمع اعداد استفاده کنید).
- ۴) بزرگترین عدد موجود در آرایه را پیدا کرده و چاپ نمایید.

نتیجه را به دستیاران آموزشی نشان دهید.

۳. انجام دهید! (مرتب سازی)

- ۱) یک فایل جدید `cpp` ایجاد کنید.
- ۲) یک آرایه به طول ۵ از جنس `int` به نام `numbers` در نظر بگیرید.
- ۳) ۵ عدد از ورودی خوانده و در آرایه بریزید.

۴) حال این عدد ها را به صورت صعودی مرتب^۴ کنید. (این کار را می توانید توسط الگوریتم Bubble Sort که قبلاً با آن آشنا شده‌اید انجام دهید.)

https://en.wikipedia.org/wiki/Bubble_sort

۵) در نهایت اعداد مرتب شده را نمایش دهید.

نتیجه را به دستیاران آموزشی نشان دهید.

۴. انجام دهید!

۱) در تابع main یک آرایه از کاراکترها (به طول ۵) به نام msg بسازید و سعی کنید به عنوان مقدار اولیه رشته ی "Hello" را در آن بریزید.

۲) به compile error تولید شده دقت کنید. چرا با وجود آن که طول رشته و طول آرایه برابر است این خطا تولید شده است؟ درباره NULL character یا '\0' تحقیق کنید.



۳) قطعه کد زیر را اجرا کنید.

```
char msg[5] = "hell";  
msg[4] = 'o';  
printf("%s", msg);
```



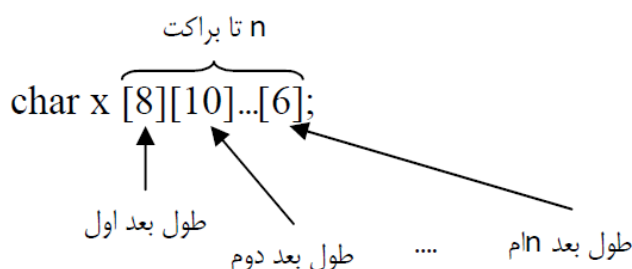
۴) چه مقداری را در خروجی مشاهده می کنید؟ سعی کنید خروجی را توجیه کنید.

۵) حال در آرایه msg رشته "BYE" را به عنوان مقدار اولیه قرار دهید و به وسیله ی یک حلقه for طول رشته ی msg را محاسبه و بر روی صفحه چاپ کنید.

نتیجه را به دستیاران آموزشی نشان دهید.

آرایه های چند بعدی:

در زبان C می توان آرایه هایی با بیش از یک بعد نیز تعریف و استفاده کرد. نحوه ی تعریف یک آرایه از نوع کاراکتر با n بعد به صورت زیر است:



دقت کنید: حافظه ی کامپیوتر مانند یک آرایه ی یک بعدی است. لذا برای شبیه سازی آرایه هایی با ابعاد بیش تر سطرهای آن را پشت سر هم قرار می دهد و با استفاده از اشاره گر به آن ها دسترسی پیدا می کند.

برای مثال آرایه ی دوبعدی char s[3][12] را در نظر بگیرید.

⁴ sort

با این کار شما سه رشته ی ۱۲ کاراکتری در اختیار دارید که می توانید با تعیین ایندکس اول به رشته ی متناظر ۱۲ تایی دسترسی داشته باشید، مثلاً `s[0]` یا `s[2]`. یا این که با تعیین کردن هر دو ایندکس به یک کاراکتر دسترسی پیدا کنید، مثلاً `s[1][7]`.

برای مقدار دهی چند بعدی می توانید از الگوی زیر استفاده کنید:

```
char s[3][12]={"Computer","IT","Electric"};
```

یا اگر آرایه از جنس `int` است:

```
int a[2][2]={{11,12},{21,22}};
```

11	12
<code>a[0][0]</code>	<code>a[0][1]</code>
21	22
<code>a[1][0]</code>	<code>a[1][1]</code>

دقت کنید که همانند آرایه های یک بعدی شما فقط هنگام تعریف یک آرایه ی چند بعدی می توانید آن را به صورت فوق مقداردهی کنید.

۵. انجام دهید!

به جای قطعه کد مشخص شده در برنامه زیر کدی قرار دهید تا در نهایت خروجی برنامه جدول ضرب باشد.

```
#include <stdio.h>
#define SIZE 9
int main() {
    int i, j;
    int table[SIZE][SIZE];۵

    Your Code Goes Here.

    for (i = 0; i < SIZE; i++) {
        for (j = 0; j < SIZE; j++)
            printf("%d ", table[i][j]);
        printf("\n");
    }
    return 0;
}
```

نتیجه را به دستیاران آموزشی نشان دهید.

^۵توجه: طول آرایه نمی تواند یک متغیر باشد. ولی می توان از یک مقدار تعریف شده استفاده کرد.