

مبانی کامپیوتر و برنامه سازی

دانشگاه آل طه

رشته کامپیوتر

درس تخصصی

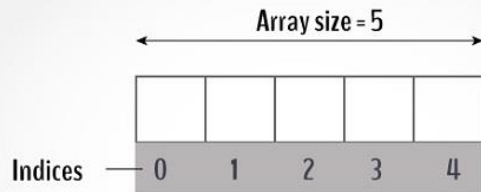
پاییز 1403

رئوس مطالب

- | | |
|--|---|
| ☐ آرایه ها | ☐ مفاهیم اولیه |
| ☐ اشاره گرها | ☐ محاسبات در کامپیوتر |
| ☐ کاراکترها و رشته ها | ☐ آشنایی با الگوریتم و فلوچارت |
| ☐ ساختارها | ☐ مقدمات برنامه سازی |
| ☐ آشنایی با ++c | ☐ فرمت بندی ورودی/خروجی |
| ☐ آشنایی با رده ها | ☐ دستورها |
| ☐ حوزه های public , private | ☐ توابع |
| ☐ پرونده ها | ☐ آزمون و خطایی برنامه |

آرایه

ساختار داده‌ای است که مجموعه‌ای از عناصر با نوع داده یکسان را تحت یک نام ذخیره می‌کند.



C++ Arrays

نوشتن آرایه

☐ `dataType arrayName[arraySize];`

☐ `dataType`: نوع آرایه

☐ `arrayName`: اسم آرایه

☐ `arraySize`: اندازه آرایه

☐ اندازه و نوع آرایه ها نمی تواند پس از تعریف تغییر کند.

☐ مثال :

`Int score[5];`

نحوه دسترسی به عناصر آرایه

فرض کنید شما آرایه mark را در بالا تعریف کردید. عنصر اول mark[0] است، عنصر دوم mark[1] و به همین ترتیب.

mark[0]	mark[1]	mark[2]	mark[3]	mark[4]

نکات کلیدی:

- خانه شروع آرایه ها ۰ است نه ۱. در این مثال mark[0] اولین عنصر است.
- اگر اندازه یک آرایه n باشد، برای دسترسی به آخرین عنصر، از (n-1) استفاده می شود. در این مثال، mark[4] آخرین عنصر است.

مقدار دهی آرایه

□ هنگام تعریف:

```
int score[5] = {25, 48, 8, 54, 7};
```

```
int score[] = {25, 48, 8, 54, 7};
```

□ بعد از تعریف:

```
score[0]=25;
```

□ دریافت همزمان چند مقدار از ورودی:

```
for (int i = 0; i < 5; ++i)
```

```
cin >> score[i];
```

نکات آرایه

در مقداردهی هنگام تعریف از خانه 0 شروع به مقدار دهی می شود و مقادیر داده نشده برابر صفر می شود.

```
Int x[5]={8,7,6}
```

اندیس 4 و 5 آرایه بالا مقدار صفر خواهند داشت.

ارسال آرایه به تابع

□ هنگامی که یک آرایه را به عنوان آرگومان به یک تابع ارسال می‌کنیم، در واقع آدرس اولین عنصر آرایه را به تابع می‌دهیم. به عبارت دیگر، تابع به جای یک کپی از آرایه، به خود آرایه اصلی دسترسی پیدا می‌کند.

```
void printArray(int arr[], int size) {  
    for (int i = 0; i < size; ++i) {  
        cout << arr[i] << " ";  
    }  
    cout << endl;  
}  
  
int main() {  
    int numbers[5] = {1, 2, 3, 4, 5};  
    printArray(numbers, 5);  
    return 0;  
}
```

نکات مهم در ارسال آرایه به تابع

□ تغییر عناصر آرایه در تابع: از آنجایی که تابع به آرایه اصلی دسترسی دارد، هر تغییری که در داخل تابع بر روی عناصر آرایه اعمال شود، بر روی آرایه اصلی نیز اعمال می‌شود.

□ اندازه آرایه: معمولاً اندازه آرایه را به عنوان یک آرگومان جداگانه به تابع ارسال می‌کنیم. زیرا کامپایلر نمی‌تواند به طور خودکار اندازه یک آرایه را که به عنوان آرگومان به تابع ارسال شده است، تعیین کند.

□ نوع ارسال آرایه به تابع از نوع ارسال با ارجاع است.

مثال

□ محاسبه مجموع عناصر آرایه

```
int sumArray(int arr[], int size) {  
    int sum = 0;  
    for (int i = 0; i < size; ++i) {  
        sum += arr[i];  
    }  
    return sum;  
}
```

روش نامرسوم

این روش در برخی کامپایلرها با خطا مواجه می شود: □

```
Int n ;
```

```
cin>>n;
```

```
Int arr[n];
```

برنامه پیدا کردن بزرگ‌ترین عدد آرایه

```
#include <iostream>

using namespace std;

int findLargest(int arr[], int size) {
    int largest = arr[0];
    for (int i = 1; i < size; i++) {
        if (arr[i] > largest) {
            largest = arr[i];
        }
    }
    return largest;
}

int main() {
    int n;
    cout << "Enter the size of the array: ";
    cin >> n;

    int arr[n];
    cout << "Enter " << n << " numbers: ";
    for (int i = 0; i < n; i++) {
        cin >> arr[i];
    }

    int largestNumber = findLargest(arr, n);
    cout << "The largest element is: " << largestNumber << endl;

    return 0;
}
```