

## عملیات آرایوی

- برای ورودی گرفتن پارامتر های دیفالت از کتابخانه `sstream` استفاده کنید
- مجاز به استفاده از کتابخانه `string` هستید
- استفاده از وکتور در این تمرین مجاز نیست

در این سوال میخواهیم یک سری توابع برای راحت تر شدن کار با آرایه یک بعدی پیاده سازی کنیم

### نکته مهم :

از بین این توابع ، سه تابع اول باید به صورت بازگشتی پیاده سازی شوند و از `current_index = start` تا `current_index = end` آرایه را به صورت بازگشتی پیمایش کنند. توجه کنید در تمام توابع پارامتر های آپشنال هستند (دیفالت پارامتر) و در صورتی که توسط کاربر مقداری برای آنها `start`, `end`, `current_index` درنظر گرفته نشود `current_index = 0` و انتهای ارایه `end = start = current_index` خواهد بود ( فقط جهت نوشتن تابع بازگشتی است و هیچگاه توسط کاربر مقدار دهی نخواهد شد ) ( باید فقط یک تابع با این اسامی پیاده سازی و فراخوانی شود )

### لیست توابع:

پیدا کردن اندیس یک عنصر داخل آرایه :  
`long long indexOf(arr, size, value, : start, end, current_index)`

این تابع اندیس مربوط به عنصر `value` را در محدوده بین اندیس `start`, `end` آرایه `arr` با سایز برمی گرداند . اگر عنصر وجود نداشته باشد مقدار -1 برگردانده می شود

شیفت به چپ :  
`void leftShift(arr, size, count, start, end, : current_index)`

این تابع عناصر در محدوده بین اندیس start, end را count واحد به چپ انتقال داده و جای count عنصر اول صفر قرار می دهد . نمونه : اگر عناصر ارایه ارایه {1, 2, 3, 4} را یک واحد به چپ شیفت بدهیم به صورت {0, 1, 2, 3} خواهد شد

```
void rightShift(arr, size, count, start, end, : راست  
current_index)
```

این تابع عناصر در محدوده بین اندیس start, end را count واحد به راست انتقال داده و جای count عنصر اول صفر قرار می دهد . نمونه : اگر عناصر ارایه ارایه {1, 2, 3, 4} را یک واحد به راست شیفت بدهیم به صورت {2, 3, 4, 0} خواهد شد

```
چرخش به چپ : void leftRotate(arr, size, count, start, end)
```

این تابع عناصر در محدوده بین اندیس start, end را count واحد به چپ چرخش میدهد. نمونه : اگر عناصر ارایه ارایه {1, 2, 3, 4} را یک واحد به چپ چرخش بدهیم به صورت {2, 3, 4, 1} خواهد شد

```
چرخش به راست : void rightRotate(arr, size, count, start, end)
```

این تابع عناصر در محدوده بین اندیس start, end را count واحد به راست چرخش میدهد. نمونه : اگر عناصر ارایه ارایه {1, 2, 3, 4} را یک واحد به راست چرخش بدهیم به صورت {4, 1, 2, 3} خواهد شد

```
مرتب سازی آرایه : void sort(arr, size, mode, start, end)
```

این تابع عناصر موجود در محدوده بین اندیس start, end را به صورت صعودی mode = 1 یا نزولی mode = 2 مرتب میکند

```
ورودی گرفتن آرایه : void input(arr, size)
```

```
چاپ یک آرایه : void print(arr, size)
```

تابع main

بعد نوشتن توابع فوق، در تابع `int main()` ابتدا یک آرایه را ورودی گرفته و بعد با استفاده از یک حلقه، منوی طراحی کنید که کاربر بتواند توابع فوق را روی آرایه خود تست کند

### گزینه های منو :

`indexOf , leftShift , rightShift , leftRotate , rightRotate , sort , exit`

در هر گزینه منو ابتدا ارگومان های غیر آپشنال توابع از کاربر ورودی گرفته شده و سپس آرگومان های آپشنال `start, end` ورودی گرفته می شوند (ممکن است کاربر مقداری برای آنها وارد نکند)

### ورودی

در بالا توضیح داده شده ( )

### خروجی

در پایان اجرای گزینه هایی از منو که روی آرایه تغییرات اعمال می شود، آرایه را چاپ کنید و در سایر گزینه ها خروجی توابع فراخوانی شده را خروجی دهید

### ورودی نمونه

```
5
8 4 6 3 9
indexOf 2
indexOf 6
indexOf 6 0 1
indexOf 6 1
rightRotate 2
leftRotate 2
rightRotate 2 2 4
leftShift 2 1
exit
```

### خروجی نمونه

-1  
2  
-1  
2  
3 9 8 4 6  
8 4 6 3 9  
8 4 3 9 6  
8 9 6 0 0