

## سورت تدریسیاران

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

تیم تدریسیاری درس الگوریتم در حین طراحی سوالات به الگوریتم جدیدی برای مرتب سازی به نام TASort دست یافتند که از نظر آنها این الگوریتم از مرتبه زمانی  $O(n)$  می باشد. شبه کد این الگوریتم در ادامه آمده است.

```
void TASort(int A, int l, int r) {  
    // Sorts [l, r)  
    if (is_sorted(a, l, r)) return;  
    int mid = (l + r) / 2;  
    TASort(A, l, mid);  
    TASort(A, mid, r);  
    merge(A, l, r);  
}
```

از آنجا که تیم تدریسیاری در محاسبه مرتبه زمانی این الگوریتم اشتباه کرده اند در این سوال باید کدی پیاده سازی کنید که به آنها نشان دهید تعداد دفعاتی که تابع TASort صدا زده می شود بیش از چیزی است که آنها فکر میکنند.

به طور دقیق تر شما یک آرایه و تعدادی پرسش از تیم تدریسیاری دریافت میکنید. به این صورت که هر پرسش یک بازه از آرایه داده شده است و به ازای هر بازه باید تعداد دفعاتی که تابع TASort صدا زده میشود را حساب کنید.

## ورودی

در خط اول ورودی دو عدد طبیعی  $n$  و  $q$  با فاصله از هم آمده است که به ترتیب طول آرایه و تعداد پرسش‌ها هستند. در سطر بعد  $n$  عدد می‌آیند که با فاصله از یک‌دیگر جدا شده و  $i$ امین عدد برابر با مقدار  $i$ ام در آرایه  $a$  می‌باشد.

$$1 \leq n, q \leq 5000$$

$$-10^8 \leq a_i \leq 10^8$$

سپس  $q$  خط در ادامه ورودی می‌آید که خط  $i$ ام از این  $q$  خط شامل دو عدد  $l_i$  و  $r_i$  می‌باشد و شما باید به ازای هر کدام به ترتیب تعداد دفعات صدا شدن تابع TASort را به ازای اجرا کردن  $\text{TASort}(a, l[i], r[i])$  حساب کرده و در خروجی چاپ کنید.

$$1 \leq l_i < r_i \leq n + 1$$

## خروجی

شامل  $q$  خط که در خط  $i$ م یک عدد صحیح که جواب پرسش متناظر می‌باشد.

## مثال

### ورودی نمونه ۱

```
10 5
6 18 27 33 2 13 12 14 22 21
1 3
2 6
5 10
5 11
1 9
```

### خروجی نمونه ۱

1  
5  
3  
11  
5

ورودی نمونه ۲

8 2  
34 7 11 27 2 35 32 16  
1 5  
3 7

خروجی نمونه ۲

5  
3