

## Funeral

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

یک آرایه به طول  $n$  از اعداد صحیح متمایز به نام  $a$  داریم.

این آرایه را با الگوریتم مرتب‌سازی سریع ( $Quick\ Sort$ ) از کوچک به بزرگ مرتب کرده‌ایم و ترتیب انتخاب  $pivot$ ها در الگوریتمی که ما پیاده کرده‌ایم به شرح زیر به شما داده شده است:

```
pivots_list = an empty list
QuickSort(A, p, r)
    if p < r
        q <- partition(A, p, r)
        pivots_list.append(A[q])
        QuickSort(A, p, q-1)
        QuickSort(A, q+1, r)
    if p == r
        pivots_list.append(A[p])
```

ترتیب انتخاب  $pivot$ ها همان ترتیب موجود در لیست pivots\_list است.

حال فرض کنید آرایه  $a$  با ترتیب pivots\_list که در ورودی به شما داده می‌شود با الگوریتم مرتب‌سازی سریع مرتب شده است. چند مقایسه در کل اجرای الگوریتم صورت گرفته است؟

فرض کنید وقتی عدد  $A[q]$  در یک قسمت از آرایه به طول  $r - p + 1$  به عنوان  $pivot$  انتخاب می‌شود باید با همه‌ی اعداد دیگر آن قسمت مقایسه شود؛ یعنی به تعداد مقایسه‌ها  $r - p$  واحد افزوده می‌شود.

### ورودی

در خط اول ورودی به شما عدد صحیح  $n$ ، طول آرایه داده شده است.

در خط دوم  $n$  عدد صحیح متمایز که با فاصله از یک‌دیگر جدا شده‌اند آمده، که اعداد آرایه‌ی  $a$  را نشان می‌دهد.

در خط سوم و پایانی ورودی  $n$  عدد صحیح که با فاصله از یک‌دیگر جدا شده‌اند آمده، که ترتیب انتخاب  $pivot$ ها ( pivots\_list ) را نشان می‌دهد. این اعداد در واقع ترتیبی دیگر از اعداد آرایه‌ی  $a$  هستند.

$$1 \leq n \leq 10^5$$

$$1 \leq a_i \leq 10^9$$

### خروجی

در تنها خط خروجی، یک عدد تعداد مقایسه‌های صورت گرفته را چاپ کنید.

### مثال

#### ورودی نمونه ۱

```
5
5 1 3 2 4
1 3 2 5 4
```

خروجی نمونه ۱

8

ورودی نمونه ۲

5  
1 27 324 415 666  
666 415 324 27 1

خروجی نمونه ۲

10