

## جایگشت مفید

- محدودیت زمان: ۲ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

احتمالا از پیش با الگوریتم **Merge Sort** آشنا هستید. امین برای اینکه این الگوریتم را کمی بهینه کند، تصمیم گرفته تنها در صورتی که آرایه از پیش مرتب‌شده نباشد، آن را به دو نیم تقسیم کرده و برای هر کدام از دو قسمت، تابع mergeSort را صدا بزند. برای مثال آرایه  $[1, 2, 3, 4, 5]$  را در نظر بگیرید. ابتدا یک بار تابع mergeSort را برای این آرایه صدا می‌کنیم. این تابع با بررسی آرایه متوجه می‌شود که این آرایه در همین حالت مرتب‌شده است و در همین نقطه، از تابع خارج می‌شود. در واقع در این حالت **فقط یک بار** این تابع صدا زده شده‌است. حال آرایه  $[3, 2, 5]$  را در نظر بگیرید. ابتدا یک بار تابع mergeSort را برای این آرایه صدا می‌زنیم. این تابع آرایه را بررسی می‌کند و متوجه می‌شود که آرایه از قبل به صورت مرتب‌شده نیست. در نتیجه، اندیس mid را به صورت زیر پیدا می‌کند :

$$mid = \left\lfloor \frac{l + r}{2} \right\rfloor = \left\lfloor \frac{0 + 3}{2} \right\rfloor = 1$$

سپس، یک بار تابع mergeSort را برای اندیس 0 تا 1 (آرایه  $[3]$ ) و بار دیگر برای اندیس 1 تا 3 (آرایه  $[2, 5]$ ) صدا می‌زند. همانطور که مثال‌ها نشان می‌دهد، آرایه‌ها شامل اندیس انتهایی نیستند. در نهایت می‌بینیم که به ازای آرایه  $[3, 2, 5]$ ، تابع mergeSort به تعداد 3 بار صدا زده شده‌است. حال امین برای توسعه الگوریتم دیگری، نیاز دارد یک جایگشت مفید با اندازه  $n$  پیدا کند؛ به طوری که اگر بخواهیم الگوریتم Merge Sort را بر روی آن اعمال کنیم، دقیقا  $m$  بار تابع mergeSort بر روی آن صدا زده شود. یک جایگشت مفید با اندازه  $n$  یک آرایه  $n$  عضوی است که شامل تمام اعداد بازه  $[1, n]$  خواهد بود.

## ورودی

ورودی تنها شامل یک خط است که در آن به ترتیب دو عدد  $n$  و  $m$  که توسط فاصله از هم جدا شده‌اند، داده می‌شود.

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

$$1 \leq m \leq 2 \times 10^5$$

## خروجی

خروجی شامل یک خط است که در آن،  $n$  عدد که توسط فاصله از هم جدا شده‌اند، که اعضای جایگشت مفید با اندازه  $n$  خواهد بود. همچنین در صورتی که چنین جایگشتی وجود ندارد، مقدار خروجی 1- خواهد بود.

## مثال

### ورودی نمونه 1

3 3

### خروجی نمونه 1

2 1 3

### ورودی نمونه 2

4 1

### خروجی نمونه 2

1 2 3 4

### ورودی نمونه 3

5 6

خروجی نمونه 3

-1

## الگوریتم مرتب‌سازی عجیب غریب

- محدودیت زمان: ۲ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

کایوت به دلایل نامعلومی تصمیم گرفته برای شکار کردن رودرانر یک الگوریتم مرتب‌سازی جدید بنویسد. اینکه این الگوریتم چگونه می‌تواند به او کمک کند، هنوز مشخص نیست و دانشمندان در حال بررسی این موضوع هستند. اما همانطور که انتظار می‌رفت، این الگوریتم به درستی کار نمی‌کند. رودرانر که از اشتباه کایوت در این الگوریتم آگاه شده، می‌خواهد جهت تمسخر او، یک آرایه به این الگوریتم بدهد و تعداد اشتباهات در خروجی الگوریتم را محاسبه کند. تعداد اشتباهات در آرایه  $A$  از به صورت تعداد جفت  $i$  و  $j$ -هایی در نظر می‌گیریم که  $i < j$  و  $A[i] > A[j]$  باشد. وظیفه شما این است که خروجی الگوریتم کایوت را گرفته و تعداد اشتباهات آن را محاسبه کنید.

### ورودی

در خط اول ورودی عدد  $n$  داده می‌شود که تعداد اعضای آرایه ورودی را نشان می‌دهد. در خط دوم ورودی  $n$  عدد که با فاصله از یکدیگر جدا شده‌اند داده شده که اعضای آرایه ورودی هستند.

$$1 \leq n \leq 200000$$

$$0 \leq A[i] \leq 10^9$$

### خروجی

خروجی تنها شامل یک عدد است که تعداد اشتباهات آرایه را نشان می‌دهد.

### مثال

## ورودی نمونه ۱

5  
1 2 4 3 5

## خروجی نمونه ۱

1

## ورودی نمونه ۲

3  
2 3 1

## خروجی نمونه ۲

2

## اسپویلر

- محدودیت زمان: ۲ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

در قسمت پنجم جان ویک در یک صحنه خاص، جان مجبور شده یک دوئل عجیب انجام دهد. در این دوئل، تعداد  $n$  اسلحه که روی هرکدام یک عدد صحیح نوشته شده (عدد روی اسلحه  $i$ -ام را  $W[i]$  می‌نامیم)، در یک ردیف و در کنار همدیگر چیده شده‌اند. برای اینکه جان بتواند در این دوئل شرکت کند، باید یک زیرمجموعه متوالی از این اسلحه‌ها را انتخاب کند و تنها با استفاده از اسلحه‌های زیرمجموعه انتخاب شده، دوئل را انجام دهد. وینستون از قبل به جان گفته است که برای برنده شدن در این دوئل، باید قدر مطلق جمع اعدادی که روی اسلحه‌های انتخاب شده نوشته شده است، از عدد ثابتی نظیر  $k$  بیشتر باشد. در واقع اگر جمع اعدادی که روی اسلحه‌های انتخاب شده توسط جان از اسلحه  $i$ -ام تا اسلحه  $j$ -ام باشد، جان در صورتی برنده دوئل خواهد بود که رابطه زیر برقرار باشد :

$$|W[i] + W[i+1] + \dots + W[j-1] + W[j]| > k$$

جان می‌داند تعداد کل حالاتی که می‌تواند از بین  $n$  اسلحه، یک زیر مجموعه متوالی انتخاب کند، از رابطه  $\frac{n(n+1)}{2}$  بدست می‌آید. اما برای اینکه احتمال برد خود را بدست آورد، نیاز دارد تعداد حالات انتخابی که شرط بالا را داشته باشد، محاسبه کند. به همین دلیل از شما کمک خواسته است. وظیفه شما در این سوال این است که اعداد روی اسلحه‌ها را دریافت کرده و تعداد حالاتی که باعث می‌شود جان برنده این دوئل باشد را پیدا کنید. همچنین با توجه به اینکه ممکن است این صحنه چند بار ضبط شود و در هر بار ضبط ترتیب اسلحه‌ها و اعداد روی آن‌ها تغییر کند، لازم است الگوریتم خود را به تعداد  $T$  بار اجرا کنید.

## ورودی

در خط اول ورودی عدد  $T$  می‌آید که نشان‌دهنده تعداد ضبط‌های این دوئل است.

سپس به ازای هر ضبط، در خط اول ابتدا عدد  $n$  و بعد عدد  $k$  و در خط بعدی  $n$  عدد که با فاصله از همدیگر جدا شده‌اند داده می‌شود به طوری که عدد  $i$ -ام نشان‌دهنده عدد نوشته‌شده روی اسلحه  $i$ -ام ( $W[i]$ ) است

$$1 \leq T \leq 1000$$

$$1 \leq n \leq 2 \times 10^5$$

$$0 \leq k \leq 10^{18}$$

$$|W[i]| \leq 10^9$$

مجموع  $n$  در تمامی ضبیطها حداکثر  $2 \times 10^5$  خواهد بود.

## خروجی

خروجی شامل  $T$  خط است که خط  $i$ -ام، تعداد حالات پیروزی جان در ضبیط  $i$ -ام را نشان می‌دهد.

## مثال

### ورودی نمونه ۱

```
2
3 2
2 -2 3
5 0
2 -3 0 3 2
```

### خروجی نمونه ۱

```
2
13
```

## دسته‌بندی

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

کانتست امسال، طبق روال معمول، قوانین عجیب و غریبی دارد. در مسابقه امسال، افراد به  $2^n$  تیم تقسیم می‌شوند به گونه‌ای که تیم  $i$ ام، به اندازه  $p_i$  قدرت حل مسئله داشته باشد. در هر مرحله از مسابقه مسئول کانتست تیم‌های حاضر را به دو دسته تقسیم می‌کند که دسته‌ی اول به ترتیب نصفه‌ی اول تیم‌های باقی‌مانده و دسته‌ی دوم نصفه‌ی دوم هستند. سپس به هر دسته، برگه‌ی سوالات داده می‌شود. مسئول تیم در انتهای این مرحله، یک دسته را حذف کرده و مرحله بعدی را با دسته دیگر ادامه می‌دهد. همچنین به پرقدرت‌ترین تیم دسته‌ی حذف‌شده، به اندازه‌ی قدرتش جایزه داده می‌شود. اسپانسر این مسابقه برای دلخور نشدن شرکت‌کنندگان می‌خواهد بیش‌ترین جایزه ممکن را طبق قوانین کانتست به شرکت‌کنندگان بدهد. برای همین، از شما کمک می‌خواهد که به او بگویید حداکثر چه مقدار پول باید از سرمایه‌گذار دریافت شود. دقت کنید که به تیم آخر هم جایزه تعلق می‌گیرد. در واقع باید در هر مرحله، تیم از دست رفته را به نحوی انتخاب کرد که مجموع جایزه‌ای که داده می‌شود، بیشینه باشد.

برای مثال، اگر ۸ تیم در کانتست شرکت کنند که قدرت‌هایشان به ترتیب برابر ۱ ۶ ۱ ۷ ۱ ۸ ۱ ۴ باشد، می‌توان در مرحله‌ی اول، ۴ تیم اول (یعنی تیم‌های به شماره ۱ و ۲ و ۳ و ۴) را حذف کرد و به اندازه ۸ تا به تیم شماره ۳ جایزه داد. سپس ۲ تیم دسته‌ی دوم (تیم‌های به شماره ۷ و ۸) را حذف کرد و ۶ تا به تیم شماره ۷ جایزه داد و در مرحله‌ی بعد، با حذف تیم شماره ۵، به مقدار ۷ تا به او جایزه داد و در انتها به تیم شماره نیز ۱ جایزه داد که در مجموع می‌شود  $1 + 7 + 6 + 8$  یعنی ۲۲ جایزه.

روش دیگر حذف تیم‌ها، حذف ۴ تیم دوم و سپس حذف تیم‌ها با شماره ۳ و ۴ و در نهایت حذف تیم اول است که در این صورت باید به اندازه  $1 + 4 + 8 + 7 = 20$  تا جایزه داده شود که در روش اول، مقدار جایزه بیشتری توزیع می‌شد.

## ورودی



در سطر اول ورودی  $n$  آمده است که نشان‌دهنده تعداد مراحل است. در خط بعدی  $2^n$  عدد آمده است که عدد  $i$ -ام نشان‌دهنده  $p_i$  است.

$$1 \leq n \leq 17$$

$$0 \leq p_i \leq 10^9$$

## خروجی

خروجی شامل یک عدد است، که نشان‌دهنده مجموع پولی است ه این مسئول باید هدیه بدهد.

## مثال

### ورودی نمونه ۱

2  
1 2 3 4

### خروجی نمونه ۱

9

ابتدا تیم‌های 1 و 2 را حذف کرده و 2 واحد جایزه داده می‌شود؛ سپس تیم 3 را حذف کرده و 3 واحد جایزه داده می‌شود و در نهایت با حذف تیم 4، 4 واحد جایزه تعلق می‌گیرد.

$$2 + 3 + 4 = 9$$

### ورودی نمونه ۲

3

1 3 5 7 2 4 6 8

خروجی نمونه ۲

25