

اولین ۱

- محدودیت زمان: ۳ ثانیه
- محدودیت حافظه: ۶۴ مگابایت

مهدی می‌خواهد در شرکت کدنویس‌گستران شرق به‌جزر نیمه استخدام شود. به‌عنوان اولین سؤال در مصاحبه، به او آرایه‌ای مرتب‌شده از ۰ ها و ۱ ها داده‌اند و به او گفته‌اند برنامه‌ای بنویسد که با دریافت آرایه، اندیس اولین ۱ موجود در آرایه را چاپ کند (اندیس‌ها از صفر شروع می‌شوند). اگر هیچ ۱ ای در آرایه موجود نبود، برنامه باید ۱- را چاپ کند.

از آن‌جا که مهدی مبانی را با نمره پایین پاس کرده است، از شما می‌خواهیم تا این برنامه را برای او بنویسید.

ورودی

در خط اول ورودی استاندارد، عدد طبیعی N نوشته می‌شود که بیانگر تعداد اعضای آرایه است.

$$1 \leq N \leq 5 \times 10^6$$

در خط دوم، N عدد با فاصله نوشته می‌شوند که اعضای آرایه هستند.

$$a_i \in \{0, 1\}$$

خروجی

در یک خط از خروجی استاندارد، اندیس اولین ۱ موجود در آرایه را چاپ کنید. اگر هیچ ۱ ای در آرایه موجود نبود، ۱- را چاپ کنید.

ورودی نمونه ۱

5

0 0 1 1 1

خروجی نمونه ۱

2

ورودی نمونه ۲

4

0 0 0 0

خروجی نمونه ۲

-1

هیچ 1 ای در آرایه موجود نیست.

اسلایس

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۵۰ مگابایت

آرایه به ساده‌ترین شکل ممکن در سی وجود دارد. همانطور که می‌دانید تنها قابلیت‌هایی که به ما می‌دهد، ساخت آرایه با اندازه معلوم (در compile time) و مقدار دادن و خواندن مقدار از اندیس حافظه است. (در سی استاندارد)

اما همه زبان‌ها اینطور نیستند. برخی زبان‌ها تحت نام‌هایی مثل list (در پایتون و جاوا) و vector (در سی پلاس پلاس) یا slice (در go) اقدام به افزایش قابلیت‌های آرایه کرده‌اند.

قابلیت‌هایی که آرایه اضافه می‌شوند متعدد اند. مهم‌ترین قابلیت‌هایی که اضافه می‌شود طول پویا است. (مثلاً بعد از ساخت آرایه به اندازه آن اضافه شود). اما این قابلیت با دانش فعلی شما قابل پیاده‌سازی نیست.

در عوض از شما می‌خواهیم قابلیت slice در لیست پایتون را به سی/سی‌پلاس‌پلاس اضافه کنید!

اسلایس یکی از قابلیت‌های خوب و دوست داشتنی پایتون است. به کمک این قابلیت می‌توان از لیستی مثل لیست زیر، خروجی‌های زیر را دریافت کرد

```
1 | a = [1,2,3,4]
2 |
3 | first = a[0]
4 | # 1
5 |
6 | last = a[-1]
7 | # 4
8 |
9 | even_poses = a[1:4:2]
10 | # 2 4
11 |
12 | first_two = a[0:2]
13 | # 1 2
14 |
```

```

15 | last_one = a[3:4]
16 | # 4
17 |
18 | all_list = a[1:4:1]
19 | # 1 2 3 4

```

انواع دستورات

همانطور که در کد پایتون نمونه می‌بینید، چند مدل دستور مختلف را می‌توان اجرا کرد. این دستورات هر کدام معنی خاصی دارند که در زیر توضیحات آن‌ها را می‌بینیم.

دستورات تک ورودی

در این حالت، یک اندیس از لیست درخواست می‌شود و آن عضو برگردانده می‌شود. اندیس‌های نامنفی دقیقا برابر اندیس‌های سی هستند. برای مثال اندیس 1 به دومین المنت از لیست اشاره دارد.

تنها نکته‌ای که وجود دارد اندیس‌های منفی هستند. در لیست پایتون معنی خاصی دارند. در صورتی که اندیس منفی باشد، مثل این است که از آخر اندیس داده‌ایم.

- اندیس 1- به معنی اندیس پیش از صفر است یعنی آخرین عنصر.
- اندیس 2- به معنی یکی قبل از آخرین عنصر است مثلا اگر لیست 4 عضو داشته باشید، اندیس‌های ۲ و ۲- به یک معنی هستند.

دقت کنید که برای یک لیست ۴ تایی، بازه‌ی معنی دار اندیس بین 4- تا 3 است.

نمونه ورودی ۱

```

1 2 3 4 5 6
1
2

```

نمونه خروجی ۱

3

در این مثال اعداد ۱ تا ۶ ورودی داده شدند. یک در خط دوم به معنی دستور تک ورودی است. در نهایت ۲ یعنی اندیس ۲ را می‌خواهیم که مقداری برابر ۳ دارد.

نمونه ورودی ۲

1 2 3 4 5 6

1

-2

نمونه خروجی ۲

5

در این مثال اعداد ۱ تا ۶ ورودی داده شدند. یک در خط دوم به معنی دستور تک ورودی است. در نهایت -2 یعنی اندیس 6-2 را می‌خواهیم که مقداری برابر 5 دارد.

دستورات دو ورودی

این دستورات یک بازه از لیست را مشخص می‌کنند. عدد اول به معنی ابتدای بازه و عدد دوم به معنی انتهای بازه است. در این حالت برنامه لیستی از اعداد با اندیس بزرگ‌تر مساوی اندیس شروع و کوچک‌تر از اندیس پایان بر می‌گرداند.

دقت کنید این اندیس‌ها نیز می‌توانن با قواعد گفته شده منفی یا مثبت باشند.

نمونه ورودی ۳

1 2 3 4 5 6

2

1 3

نمونه خروجی ۳

2 3

در این مثال اعداد ۱ تا ۶ ورودی داده شدند. دو در خط دوم به معنی دستور دو ورودی است. در نهایت 1 3 یعنی از اندیس ۱ تا کوچکتر از ۳ (اندیس ۱ و ۲) را می‌خواهیم.

نمونه ورودی ۴

1 2 3 4 5 6

2

1 -1

نمونه خروجی ۴

2 3 4 5

در این مثال اعداد ۱ تا ۶ ورودی داده شدند. دو در خط دوم به معنی دستور دو ورودی است. در نهایت 1- 1 یعنی از اندیس ۱ تا قبل از اندیس آخر را می‌خواهیم.

نمونه ورودی ۵

1 2 3 4 5 6

2

0 6

نمونه خروجی ۵

1 2 3 4 5 6

در این مثال اعداد ۱ تا ۶ ورودی داده شدند. دو در خط دوم به معنی دستور دو ورودی است. در نهایت ۱ تا قبل از ۶ از ما خواسته شده که یعنی کل لیست. در اینجا اگرچه خود 6 اندیس ولیدی در لیست نیست ولی

به عنوان پایان مورد قبول است.

دستورات ۳ ورودی

در نهایت گونه‌ای از دستورات داریم که ۳ ورودی می‌گیرند. ۲ ورودی اول مانند قسمت قبل هستند. اما تاثیر ورودی سوم این است که می‌گوید در هر گام چند تا جلو برو. در حالت عادی (دستور دو ورودی) گام برابر ۱ است

نمونه ورودی ۶

1 2 3 4 5 6
3
1 -1 1

نمونه خروجی ۶

2 3 4 5

این مثال دقیقا مثل مثال ۴ عمل می‌کند. از ۱ تا یکی مانده به آخر را «یکی یکی» چاپ می‌کند.

نمونه ورودی ۷

1 2 3 4 5 6
3
1 6 4

نمونه خروجی ۷

2 6

با شروع از ۱، ۴ تا ۴ تا جلو می‌رویم. اندیس بعدی ۵ است. بنابراین اندیس‌های ۱ و ۵ چاپ می‌شوند. اگر ۴ تا جلو برویم خارج از لیست است پس ادامه نمی‌دهیم.

توجه: در پایتون، ورودی سوم (گام) نیز می‌تواند منفی باشد ولی در این سوال از شما نخواسته‌ایم.

ورودی

در اولین خط ورودی ۶ عدد پشت سر هم می‌آیند. این ۶ عدد اعداد لیست شما هستند.

در خط بعدی یک عدد می‌آید که تعداد ورودی دستور است. (مشخص کننده دستور یک ورودی یا دو ورودی یا ۳ ورودی)

در خط سوم، ورودی‌های دستور می‌آیند.

تضمین

- آرایه‌ها حتما ۶ عضوی هستند.
- اندیس‌ها طبق قواعد گفته شده معتبر هستند.
- بازه اعداد لیست، بین ۰ تا ۱۰ است.

خروجی

در یک خط، لیستی که حاصل محاسباتتان است را مانند مثال‌ها چاپ کنید.

سمت Rust

- محدودیت زمان: ۵ ثانیه
- محدودیت حافظه: ۶۴ مگابایت

اخیراً نمرات امتحان میان‌ترم نظریه‌ی زبان‌ها و ماشین‌ها منتشر شده است. TAهای این درس این نمرات را به ترتیب شماره‌ی دانشجویی افراد (صعودی) نوشته‌اند و آن را در کانال این درس قرار داده‌اند. حال، دانشجویان این درس می‌خواهند بدانند که نمره‌ی کدام افراد از نمره‌ی تمامی افرادی که شماره‌ی دانشجویی‌شان بزرگ‌تر از آن‌هاست بیش‌تر یا مساوی است. در واقع، می‌خواهند نمراتی را بدانند که از نمرات سمت راست‌شان در لیست بزرگ‌تر یا مساوی باشد. برنامه‌ای بنویسید که پاسخ سؤال آن‌ها را محاسبه کند.

ورودی

در خط اول ورودی استاندارد، عدد طبیعی n وارد می‌شود که بیانگر تعداد نمرات است.

$$1 \leq n \leq 10^6$$

در خط دوم، نمرات افراد به‌ترتیب وارد می‌شود. این نمرات به‌صورت عدد صحیح بوده و حداکثر مقدار آن 10^5 است!

$$0 \leq g_i \leq 10^5$$

خروجی

در یک خط از خروجی استاندارد، نمراتی که از نمرات سمت راست‌شان در لیست بزرگ‌تر یا مساوی هستند را با فاصله چاپ کنید.

ورودی نمونه ۱

6

16 17 4 3 5 2

خروجی نمونه ۱

17 5 2

ورودی نمونه ۲

8

456 132 987 456 213 654 3212 23

خروجی نمونه ۲

3212 23

راهنمایی: برای تعریف آرایه، یک آرایه با طول ثابت تعریف کنید که به ازای هر n مناسب باشد.

عضو مکرر

- محدودیت زمان: ۵ ثانیه
- محدودیت حافظه: ۶۴ مگابایت

آرایه‌ی a شامل n عدد حسابی است. برنامه‌ای بنویسید که با دریافت اعضای آرایه‌ی a و عدد طبیعی k ، کوچک‌ترین عددی که دقیقاً k بار تکرار شده است را چاپ کنید. تضمین می‌شود حداقل یک عدد وجود دارد که دقیقاً k بار تکرار شده است.

ورودی

در خط اول از ورودی استاندارد، عدد طبیعی N نوشته می‌شود که بیانگر تعداد اعداد موجود در آرایه‌ی a است.

$$1 \leq N \leq 10^5$$

در خط دوم، N عدد حسابی با فاصله نوشته می‌شوند که اعداد موجود در آرایه‌ی a هستند.

$$0 \leq a_i \leq 10^5$$

در خط سوم، عدد طبیعی k نوشته می‌شود.

$$1 \leq k \leq 10^5$$

خروجی

در یک خط از خروجی استاندارد، کوچک‌ترین عددی که دقیقاً k بار تکرار شده است را چاپ کنید.

ورودی نمونه ۱

5
2 2 1 3 1
2

خروجی نمونه ۱

1

ورودی نمونه ۲

8
5 5 8 4 4 9 1 6
2

خروجی نمونه ۲

4

راهنمایی: همیشه لازم نیست خود ورودی‌ها را در آرایه بریزیم. می‌توان از ورودی به عنوان اندیس استفاده کرد.

مار و پله

در این سوال از شما می‌خواهیم که بازی محبوب مار و پله (به انگلیسی snakes and ladders) را پیاده‌سازی کنید.

برای نمونه به تصویر زیر نگاه کنید:



به چینش ردیف‌ها دقت کنید. در بازی مورد نظر هم همین چینش سطر ها مورد انتظار است.

برای ساخت زمین در ابتدا، تعدادی مار و پله در ورودی به شما داده می‌شود. مار و پله‌ها با شماره مبدا و مقصدشان شروع می‌شوند. برای مثال یک مار می‌تواند شروعی برابر ۹۰ و پایانی برابر ۵ داشته باشد. و یا پله می‌تواند از ۷ شروع شود و به ۵۰ برسد. (اندازه زمین مورد نظر ما ۱۰۰ است، یعنی ۱۰ ردیف ۱۰ تایی)

در ادامه تعدادی تاس در ورودی داده می‌شود. هر تاس عددی صحیح بین ۱ تا ۶ است.

در بازی ۲ بازیکن داریم که با هم به رقابت می‌پردازند. شروع آن‌ها از خانه ۱ است و با هر تاس مقداری به جلو می‌روند. (بسته به اینکه نوبت کدام باشد).

فرض می‌کنیم دنباله ورودی تاس‌ها بی‌نهایت است (هرچند واقعا اینگونه نیست!). بنابراین تا زمانی ورودی می‌خوانیم که یکی از بازیکن‌ها برنده شود و پیام مناسب چاپ شود.

برای سادگی فرض کنید که

- اندازه بازی همواره ۱۰ در ۱۰ است.
- نقطه شروع و پایان مارها و نردبان‌ها تداخلی ندارد.
- تاس‌ها هیچ جایزه‌ای ندارند بنابراین دقیقاً بعد از هر بازیکن نوبت بازیکن دیگر است.
- مجموع تعداد مار و پله‌ها از ۵۰ بیشتر نیست.
- همه مار و پله‌ها اعداد درست و داخل صفحه دارند.

توجه

- در صورتی که تاس بزرگی بیاید که منجر به خارج شده از صفحه بشود، هیچ حرکتی صورت نمی‌گیرد و نوبت بازیکن از دست می‌رود. برای مثال اگر در خانه ۹۸ است، فقط با ۱ و ۲ می‌تواند حرکت کند و اگر ۳ یا بالاتر بیاورد حرکتی نمی‌کند و نوبتش از دست می‌رود.
- شروع حرکت از خانه ۱ است.
- تضمین می‌شود بازی به سرانجام می‌رسد.
- در صورتی که نمره‌ای پایین‌تر از ۹۰ بگیرید، نمره ۰ برایتان منظور می‌شود.
- چاپ `\n` در آخر خروجی فراموش نشود.

ورودی

در اولین خط ورودی تعداد پرش‌ها (مار یا پله‌ها) داده می‌شود (n). در هر یک از n خط بعدی، ۲ عدد می‌آید که به ترتیب شروع و پایان یک پرش است. در هر یک از بی‌نهایت خط بعدی (!) یک عدد صحیح ۱ تا ۶ به عنوان تاس می‌آید.

خروجی

در صورتی که بازیکن اول برنده می‌شود عبارت `player1 win` و در صورتی که بازیکن دوم برنده می‌شود، عبارت `player2 win` را چاپ کنید.

مثال

ورودی نمونه ۱

```
1
2 98
1
5
2
// continue ..
```

خروجی نمونه ۱

```
player1 win
```

توضیح

۱. بازیکن ۱، با تاس ۱ به خانه ۲ (و سپس ۹۸) رفت.
۲. بازیکن ۲، با تاس ۵ به خانه ۶ رفت.
۳. بازیکن ۱ با تاس ۲ از خانه ۹۸ به ۱۰۰ رفته و بازی را برد.

ورودی نمونه ۲

```

3
2 48
51 30
4 100
1
2
3
1
// continue

```

خروجی نمونه ۲

```
player2 win
```

توضیح

۱. بازیکن ۱، با تاس ۱ به خانه ۲ (و سپس ۴۸) رفت.
۲. بازیکن ۲، با تاس ۲، به خانه ۳ رفت.
۳. بازیکن ۱، با تاس ۳ به خانه ۵۱ (و سپس ۳۰) رفت.
۴. بازیکن ۲، با تاس ۱ از خانه ۳ به ۴ (و سپس ۱۰۰) رفت و بازی را برد.

راهنمایی: آرایه دو بعدی کار را سخت می‌کند.

ویرایش: فرض کنید در خانه‌ی ۱ پله وجود ندارد یا اگر وجود دارد آن را نادیده بگیرید. (بازیکن از آن بالا نمی‌رود).

نیما در استیکرفروشی پارسا (امتیازی)

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۶۴ مگابایت

متأسفانه پیشنهادی که سایت **فلربو** به نیما داده بود توسط او رد شد، زیرا قیمت استیکرهای این سایت از نظر نیما گزاف بودند. پس از بررسی سایت‌های مختلف، نیما متوجه شد که قیمت استیکرها در همه‌جا یکسان است و مشکل از توان مالی اوست. پس از این واقعه، نیما تبدیل به فردی بی‌خیال شد و ظاهر استیکرها برای او کاملاً بی‌اهمیت شد، به‌طوری که او هم‌اکنون حتی حاضر است استیکرهای **Ben 10** را روی لپ‌تاپش بچسباند. او اخیراً متوجه شده که دوست صمیمی‌اش، پارسا، یک استیکرفروشی افتتاح کرده است که در آن استیکرهای جذاب و متنوعی می‌فروشد. نیما قیمت استیکرهای پارسا را برحسب دلار به‌دست آورده و هم‌اکنون X دلار پول دارد. او می‌خواهد بداند با استفاده از پولش حداکثر چند استیکر می‌تواند از پارسا بخرد.

ورودی

در خط اول ورودی استاندارد، عدد طبیعی n وارد می‌شود که بیانگر تعداد استیکرهای موجود در استیکرفروشی پارسا است.

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

در خط دوم، n عدد طبیعی با فاصله از هم وارد می‌شوند، به‌طوری که p_i بیانگر قیمت استیکر i ام است.

$$1 \leq p_i \leq 10^9$$

در خط سوم، عدد حسابی X وارد می‌شود که بیانگر بودجه‌ی فعلی نیما است.

$$0 \leq X \leq 10^{14}$$

خروجی

در یک خط از خروجی استاندارد، حداکثر تعداد استیکرهایی که نیما با پولش می‌تواند بخرد را چاپ کنید.

ورودی نمونه ۱

5
8 6 7 2 1
12

خروجی نمونه ۱

3

ورودی نمونه ۲

7
45 98 12 35 65 99 98
57

خروجی نمونه ۲

2

- راهنمایی: از مرتب‌سازی استفاده کنید.
- استفاده از سورتِ خود زبان، مجاز نیست.
- برای حل این سوال از مرج سورت استفاده کنید، سورت‌های بابل و .. به اندازه کافی سریع نیستند.

مجموع صفر (امتیازی)

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۶۴ مگابایت

مجموعه‌ای شامل n عدد حسابی داریم. این اعداد بیانگر تعدادی زاویه هستند. برنامه‌ای بنویسید که با دریافت این زوایا از ورودی، مشخص کند که آیا می‌توان علامت‌هایی (مثبت یا منفی) به این اعداد نسبت داد به‌طوری که مجموع‌شان صفر شود یا خیر. توجه کنید که زاویه‌ی ۳۶۰ درجه نیز معادل صفر درجه است.

ورودی

در اولین خط از ورودی استاندارد، عدد طبیعی n نوشته می‌شود که بیانگر تعداد زوایا است.

$$1 \leq n \leq 15$$

در دومین خط از ورودی استاندارد، n عدد حسابی با فاصله نوشته می‌شوند که بیانگر زوایا هستند.

$$0 \leq a_i \leq 360$$

خروجی

در یک خط از خروجی استاندارد، در صورتی که می‌توان علامت‌ها را طوری به اعداد نسبت داد که مجموع‌شان صفر شود، YES و در غیر این‌صورت، NO را چاپ کنید.

ورودی نمونه ۱

3

10 20 30

خروجی نمونه ۱

YES

$$+10 + 20 - 30 = 0$$

ورودی نمونه ۲

3

360 360 360

خروجی نمونه ۲

YES

$$-360 + 360 + 360 = 360 \equiv 0$$

ورودی نمونه ۳

3

1 3 5

خروجی نمونه ۳

NO

به هیچ طریقی نمی‌توان علامت‌ها را طوری به اعداد ۱، ۳ و ۵ نسبت داد که مجموع‌شان صفر شود.