

خیالم راحت باشه؟

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

آقای محمدی که حساسیت خاصی بر ساعت خروج کارکنان دارد، مدیرعامل شرکت اندیشه‌سازان میانسال نیز است. او برای بهبود عملکرد کارکنان شرکت می‌خواهد نرم‌افزاری بنویسد تا بتواند ورود، خروج و جلسات کارکنان را مدیریت کند. در این شرکت، ساعت ورود و خروج نه صبح تا شش عصر است. تیم نرم‌افزار این شرکت موارد مربوط به ورود و خروج را انجام داده، ولی برای مدیریت جلسات از شما کمک می‌خواهد.

دقت کنید این سوال سه بخشی است و سوالات دوم و سوم در ادامه خواهد آمد.

نکته اول: دقت کنید ساعت نه صبح، برابر با epoch 32400000 و ساعت شش عصر برابر با epoch 32400000 است. نکته دوم: واحد epoch میلی ثانیه است.

در ابتدا باید یک slot خالی برای جلسه پیدا کنیم. در این قسمت، به تابع زیر مدت زمان جلسه داده می‌شود و از کاربران خواسته می‌شود تا اولین جای خالی را برای جلسه ارائه کند.

ورودی

در اولین خط ورودی، تنها یک عدد است که تعداد جلسات کارمند را نشان می‌دهد.

n

سپس در n خط بعدی، اطلاعات مربوط به جلسات 0م تا $n-1$ م می‌آید.

i start_epoch end_epoch

Copy

در خط بعدی، مشخصات جلسه‌ای گرفته می‌شود که می‌خواهیم در تقویم کارمند قرار دهیم.

n duration

Copy

خروجی

خروجی برنامه‌ی شما باید شامل یک خط باشد که در زمان شروع و پایان قرار داده می‌شود.

start_epoch end_epoch

Copy

مثال

ورودی نمونه ۱

7

0 3676 37812

1 53091 64471

2 96052 109623

3 110652 132885

4 138052 165419

5 180586 202602

6 203473 236618

7 13879

Copy

در تقویم کارمند هفت جلسه قرار داده شده است که به ترتیب آمده‌اند. و در خط هشتم، مشخصات جلسه‌ای که می‌خواهیم در تقویم قرار بدهیم آمده است.

خروجی نمونه ۱

37813 51692

Copy

بنا به تقویم کاربر جلسه باید در این epoch از روز برگزار شود.

آقای محمدی برزخ می‌شود.

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

دقت کنید این سوال ادامه سوال قبلی است.

آقای محمدی زمانی که کارمندان جلسات زیادی برگزار می‌کنند برزخ می‌شود. حالا آقای محمدی از شما می‌خواهد کدی بنویسد که میزان ساعت جلسات مشترک هر دو کارمند خود را پیدا کند. تابعی بنویسید که این کار را انجام دهد.

```
def findHoursOfCommonMeetings:  
    ## pass
```

[Copy](#)

ورودی

برای ورودی در این سوال می‌بایست اطلاعات دو کاربر را گرفت. به همین منظور در ابتدا یک عدد طبیعی n گرفته می‌شود که میزان جلسات نفر اول است. سپس در n خط بعدی، اطلاعات مربوط به جلسات 0 تا $n-1$ م می‌آید.

```
i    start_epoch    end_epoch
```

[Copy](#)

در خط بعدی، یک عدد طبیعی m گرفته می‌شود که تعداد جلسات نفر دوم است. سپس در m خط بعدی، اطلاعات مربوط به جلسات 0 تا $m-1$ م می‌آید.

```
i    start_epoch    end_epoch
```

[Copy](#)

خروجی

خروجی برنامه‌ی شما باید شامل یک خط است که تعداد epoch مشترک بین دو کارمند است.

مثال

در اینجا چند نمونه برای فهم بهتر صورت سوال و قالب ورودی و خروجی تست‌ها داده می‌شود.

ورودی نمونه ۱

6

Copy

0 29581 36017

1 63462 78043

2 110302 120093

3 145840 176662

4 186877 197551

5 232219 245789

3

0 29152 51690

1 52628 73503

2 82456 105634

کارمند اول ۶ جلسه و کارمند دوم ۳ جلسه دارد. و اطلاعات جلسات در ادامه آمده است.

خروجی نمونه ۱

16477

Copy

تعداد epoch مشترک دو کارمند 16477 است.

تقویم هوشمند

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

دقت کنید که این سوال ادامه سوالات قبلی است.

کاربران می‌توانند یک یا چند روز در هفته را روز تماماً کاری اعلام کنند و در آن روزها جلسه‌ای نگذارند. حالا، آقای محمدی برای سوال سوم از این نرم‌افزار می‌خواهد زمان مشترکی برای برگزاری جلسه بین دو یا چند کاربر را پیدا کند. دقت کنید جلسات باید در روزی باشد که برای همه‌ی کاربران روز تماماً کاری نباشد.

ورودی

در خط اول ورودی تعداد کارمندان داده می‌شود که یک عدد طبیعی n است. سپس برای n کاربر روند زیر ادامه پیدا می‌کند.

```
user_id
busy_day
number_of_meetings
for i = 0 ; i < number_of_meetings
    meeting_id day_of_week start_epoch end_epoch
```

[Copy](#)

بعد از تمامی این خط‌ها، اطلاعات مربوط به جلسه‌ی درخواستی داده می‌شود

```
meeting_id duration
```

[Copy](#)

خروجی

خروجی برنامه‌ی شما باید شامل یک خط باشد که شامل روز برگزاری جلسه، epoch شروع و epoch پایان است.

day_of_week start_epoch end_epoch

Copy

مثال

ورودی نمونه ۱

```
3
0
THURSDAY
8
0 SATURDAY 13751 19239
1 SUNDAY 40667 61453
2 SATURDAY 88740 92547
3 SATURDAY 120243 153659
4 SUNDAY 182110 215886
5 FRIDAY 235581 249694
6 TUESDAY 273514 298595
7 FRIDAY 307952 320658
1
FRIDAY
3
0 TUESDAY 16423 29404
1 MONDAY 61138 70147
2 TUESDAY 85606 91808
2
MONDAY
3
0 WEDNESDAY 3020 13903
1 THURSDAY 43654 45165
2 FRIDAY 68613 70909
9 2542
```

Copy

خروجی نمونه ۱

TUESDAY 0 2542

Copy

تقویم هوشمند

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

دقت کنید که این سوال ادامه سوالات قبلی است.

کاربران می‌توانند یک یا چند روز در هفته را روز تماماً کاری اعلام کنند و در آن روزها جلسه‌ای نگذارند. حالا، آقای محمدی برای سوال سوم از این نرم‌افزار می‌خواهد زمان مشترکی برای برگزاری جلسه بین دو یا چند کاربر را پیدا کند. دقت کنید جلسات باید در روزی باشد که برای همه‌ی کاربران روز تماماً کاری نباشد.

ورودی

در خط اول ورودی تعداد کارمندان داده می‌شود که یک عدد طبیعی n است. سپس برای n کاربر روند زیر ادامه پیدا می‌کند.

```
user_id
busy_day
number_of_meetings
for i = 0 ; i < number_of_meetings
    meeting_id day_of_week start_epoch end_epoch
```

[Copy](#)

بعد از تمامی این خط‌ها، اطلاعات مربوط به جلسه‌ی درخواستی داده می‌شود

```
meeting_id duration
```

[Copy](#)

خروجی

خروجی برنامه‌ی شما باید شامل یک خط باشد که شامل روز برگزاری جلسه، epoch شروع و epoch پایان است.

day_of_week start_epoch end_epoch

Copy

مثال

ورودی نمونه ۱

```
3
0
THURSDAY
8
0 SATURDAY 13751 19239
1 SUNDAY 40667 61453
2 SATURDAY 88740 92547
3 SATURDAY 120243 153659
4 SUNDAY 182110 215886
5 FRIDAY 235581 249694
6 TUESDAY 273514 298595
7 FRIDAY 307952 320658
1
FRIDAY
3
0 TUESDAY 16423 29404
1 MONDAY 61138 70147
2 TUESDAY 85606 91808
2
MONDAY
3
0 WEDNESDAY 3020 13903
1 THURSDAY 43654 45165
2 FRIDAY 68613 70909
9 2542
```

Copy

خروجی نمونه ۱

TUESDAY 0 2542

Copy

