

## این ک خیلی سادس

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

اصغر می خواهد یک دنباله‌ی متوازن از پرانتزها را وارد رایانه‌ی جدید خود کند، اما او که به دلیل حساسیت پوستی به هوای آلوده تهران، همواره مجبور است از دستکش استفاده کند، در هنگام استفاده از کیبورد و تایپ کردن این دنباله مشکل دارد و گاهی پرانتزها را جابه‌جا و یا اشتباه وارد می‌کند. قرار است به او کمک کنید تا حداقل تعداد پرانتزهایی را که باید تغییر کنند تا دنباله دوباره متوازن شود بیابد.

منظور از تغییر یک پرانتز عوض کردن ( ) یا برعکس است.

دنباله‌ی متوازن به دنباله‌ای گفته می‌شود که تعداد ( ها و ) ها در آن برابر باشد. همچنین در هر پیشوند از این دنباله، تعداد ( ها حداقل به اندازه‌ی تعداد ) ها باشد.

## ورودی

یک رشته از پرانتزها به طول زوج، به طوری که طول آن حداکثر  $10^5$  کاراکتر است.

## خروجی

در تنها سطر از خروجی، حقل تعداد پرانتزهایی که باید تغییر کنند تا دنباله متوازن شود را چاپ کنید.

**در حل این سوال باید به صورت پویا حافظه بگیرید و اگر حتی یک خانه اضافه تر گرفته بودید نمره ی شما ثلث میشود (:**

راهنمایی : از داینامیک الوکیشن بهره ببرید `realloc & malloc | calloc`

راهنمایی ۲: باتوجه به مطالبی که درباره ی استک میدانید (سرچ میکنید) سعی کنید استکی را به این روش پیاده سازی کنید (: انجام این کار حکم امتیاز اضافه دارد

## مثال

ورودی نمونه ۱

(()(

خروجی نمونه ۱

2

ورودی نمونه ۲

))(((((())(())(())(())

خروجی نمونه ۲

1

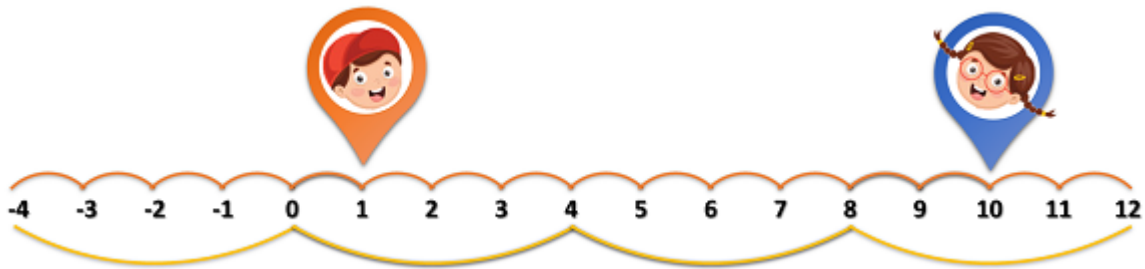
## قطار کامیابی

- محدودیت زمان: ۰.۵ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

بعد از مدت‌ها که فرهاد توانست با موفقیت چالش خوش‌اندام شدن را تا حدودی پشت سر بگذارد، شیرین نیز کم‌کم به او علاقه‌مند شد و امروز آن‌ها می‌خواهند به اولین قرار خود بروند. فرهاد در نقطه  $a$  شهر زندگی می‌کند و شیرین نقطه  $b$  را برای اولین قرار انتخاب کرده‌است.

در این شهر دو نوع قطار برای جابه‌جایی وجود دارد:

- نوع اول: نقطه  $x$  و  $x + 1$  را با یک مسیر دو طرفه به هم متصل می‌کند. (به ازای هر  $x$  صحیح)
- نوع دوم: نقطه  $k \times x$  و  $k \times (x + 1)$  را با یک مسیر دو طرفه به هم متصل می‌کند. (به ازای هر  $x$  صحیح و  $k$  داده شده در ورودی)



هم چنین می‌دانیم فاصله طی کردن یک مسیر بین دو نقطه به ازای هر نوع قطار دقیقاً یک دقیقه است. وظیفه شما به عنوان دوست و رفیق فرهاد این است که به او بگویید زودترین زمان ممکن رسیدن فرهاد به محل قرار چقدر است.

## ورودی

ورودی تنها شامل یک سطر است که در آن به ترتیب سه عدد صحیح  $k$  و  $a$  و  $b$  با فاصله از هم آمده‌است.

$$-10^9 \leq a, b \leq 10^9$$

$$1 \leq k \leq 10^9$$

## خروجی

در تنها سطر خروجی زودترین زمان رسیدن فرهاد به محل قرار را چاپ کنید.

## ورودی نمونه

4 1 10

## خروجی نمونه

5

نمونه‌ی بالا همان تصویر موجود در صورت سوال است؛ مسیر بهینه با ۵ سفر در تصویر پررنگ شده است.

## مسیر اول

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

ماجرای این قرار است که امیر حسین اعداد اول را خیلی دوست دارد... وی در خانه‌ی  $(a, b)$  یک جدول  $n \times n$  قرار گرفته و می‌خواهد به خانه‌ی  $(x, y)$  برود. از شما کمک می‌خواهد تا یک مسیر مناسب از نقطه‌ی شروع به نقطه‌ی هدف را به او نشان دهید. این مسیر باید شرایط زیر را داشته باشد:

- داخل هر خانه از جدول، عددی نوشته شده است و امیرحسین فقط از خانه‌ای عبور می‌کند که عدد روی آن اول باشد.
- هنگامی که امیرحسین روی یکی از خانه‌های جدول ایستاده است، در حرکت بعدی فقط می‌تواند به یکی از خانه‌های مجاور ضلعی‌اش برود. (بالا، پایین، چپ یا راست)
- در مسیری که وی طی می‌کند، نباید هیچ یک از خانه‌های جدول را دوبار ملاقات کند.

## ورودی

ابتدا عدد  $t$  که نشان دهنده‌ی تعداد تست‌های سوال است داده می‌شود.

$$1 \leq t \leq 100$$

به ازای هر تست: عدد  $n$  به عنوان سائز ماتریس و سپس  $n^2$  عدد که نشان دهنده‌ی مقادیر خانه‌های ماتریس هستند وارد می‌شوند.

$$1 \leq n \leq 100$$

$$1 \leq a_{i,j} \leq 10^9$$

سپس در یک خط، مختصات شروع و در خط بعدی مختصات پایان داده می‌شود. (برای فهم بهتر سوال، حتماً مثال را ببینید.)

تضمین می‌شود:

- محتویات خانه‌ی  $(a, b)$  و خانه‌ی  $(x, y)$  حتما اعدادی اول هستند.
- جواب سوال یکتاست (دو مسیر مناسب وجود نداشته باشد).
- خانه‌ی شروع حداکثر یک همسایه‌ی اول دارد و هر یک از خانه‌های مسیر حداکثر دو همسایه‌ی اول دارند. (سعی کنید به عنوان تمرین بیشتر، سوال را در حالتی حل کنید که مسیرهای انحرافی نیز وجود داشته باشند).

## خروجی

در  $t$  خط به ازای هر تست، اگر مسیر مناسبی با شرایط گفته شده از نقطه‌ی آغاز به نقطه‌ی هدف وجود دارد، آن مسیر را با کاراکترهای  $U, D, L, R$  نشان دهید و در غیر اینصورت "No Monaseb Masir!" را چاپ کنید.

برای حل این سوال باید برای هر جدول به صورت داینامیک حافظه بگیرید (2D allocation memory) و پس از چاپ کردن جواب هر تست تمام خانه‌هایی که گرفته اید را آزاد کنید (( از دستور free بهره بگیرید)).

کد شما به صورت دستی بازبینی میشود و اگر این کار را انجام نداده باشید، نمره‌ی شما در عدد منفی یک ضرب خواهد شد (:

## مثال

ورودی نمونه

3

4

5 2 6 11

4 5 13 8

7 10 3 7

2 13 15 11

0 0

3 3

3

5 2 7

8 4 1

7 5 3

2 0

0 1

6

15 2 3 5 11 77

14 5 1 4 2 1

12 7 13 6 2 4

6 8 76 9 7 95

5 11 13 7 5 8

2 96 10 4 100 17

2 2

5 0

خروجی نمونه

RDRDRD

No Monaseb Masir!

LUURRRDDDDLLLLD

## خرما

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

یک روز یک خری متعلق به مناطق بیابانی به کتابخانه رفت و به دلیل خراش (خر بودن) به جای مطالعه تصمیم گرفت کتاب ها را گاز بزند. کتابخانه  $n$  کتاب دارد که بر حسب موضوعشان با  $n$  عدد صحیح متفاوت بین 1 تا  $n$  شماره گذاری شده اند. او می خواهد همه کتاب ها را گاز بزند اما می خواهد به ترتیبی به گاز زدن بپردازد که حوصله اش سر نرود. از آنجا که هر چه عدد دو کتاب به هم نزدیک تر باشد موضوعاتشان بیشتر شبیه به هم است، در صورتی که خر ما دو کتاب متوالی گاز بزند که اعدادشان کمتر از  $k$  تا اختلاف دارند، به دلیل شباهت محتوا حوصله اش سر می رود. او خر است و از شما می خواهد که ترتیب مناسبی برای گاز زدن به او پیشنهاد دهید.

## ورودی

در تنها سطر ورودی به ترتیب دو عدد صحیح  $n$  و  $k$  می آیند که تعداد کتاب ها و عدد مربوط به حوصله خر را نشان می دهند.

$$1 \leq k \leq n \leq 1\,000\,000$$

## خروجی

در تنها خط خروجی باید ترتیب مناسبی از کتاب ها برای این که خر ما گاز بزند چاپ شود و اگر چنین ترتیبی وجود نداشت عبارت "Impossible" چاپ شود.

اگر میخواهید از آرایه استفاده کنید با توجه به ورودی های سوال باید به صورت داینامیک حافظه بگیرید و هیچ خانه ی اضافی و بی مصرفی تحت هیچ شرایطی نباید در برنامه ی شما باشد. **وگرنه نمره ی شما صفر خواهد شد**

## مثال



## ورودی نمونه ۱

5 2

## خروجی نمونه ۱

1 4 2 5 3

اگر خر ما کتاب ها را به این ترتیب گاز بزند اختلاف اعداد کتاب های متوالی به ترتیب ۳، ۲، ۳ و ۲ می شود و بنابراین اعداد کتاب های متوالی حداقل ۲ تا اختلاف خواهند داشت.

## ورودی نمونه ۲

2 2

## خروجی نمونه ۲

Impossible

در این صورت چه کتاب ها به ترتیب [1, 2] و چه به ترتیب [1, 2] گاز زده شوند، اختلاف اعداد کتاب اول و دوم کمتر از ۲ می شود.

## ایزی لیگ

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

علی به تازگی با برنامه نویسی آشنا شده است و برای همین ذوق خاصی دارد. از این رو می‌خواهد سیستمی طراحی کند که یک لیگ فوتبال را شبیه سازی می‌کند.

در این سیستم، دستورات مختلفی وجود دارند که به شرح زیر می‌باشند:

## ساختن بازیکن جدید

new player name price speed finishing defence

این دستور، بازیکنی با نام name، قیمت price، سرعت speed، قدرت تمام‌کنندگی finishing و دفاع defence می‌سازد.

توجه کنید که name از جنس رشته، و باقی پارامترها از جنس عدد می‌باشند.

هر بازیکن، یک شناسه‌ی عددی دارد که با استفاده از آن می‌توان به آن بازیکن دسترسی داشت، اولین بازیکنی که ساخته می‌شود شناسه‌ی عددی ۱، دومین بازیکنی که ساخته می‌شود شناسه‌ی عددی ۲ و i-امین بازیکنی که ساخته می‌شود شناسه‌ی عددی i دارد.

## ساختن تیم جدید

new team name money

با استفاده از این دستور، می‌توانید یک تیم جدید با نام name و پول money بسازید. در صورتی که از قبل تیمی با اسم name وجود داشته باشد، تیم جدیدی ساخته نمی‌شود.

همچنین name از جنس رشته و money از جنس عدد می‌باشد.

هر تیم، یک شناسه‌ی عددی دارد که با استفاده از آن می‌توان به آن تیم دسترسی داشت، اولین تیمی که ساخته می‌شود شناسه‌ی عددی ۱، دومین تیمی که ساخته می‌شود شناسه‌ی عددی ۲ و ...-امین تیمی که ساخته می‌شود شناسه‌ی عددی  $i$  دارد.

## خرید بازیکن توسط تیم

buy playerID teamID

با استفاده از این دستور، تیم با شناسه‌ی عددی teamID بازیکن با شناسه‌ی عددی playerID را می‌خرد.

همچنین همانطور که در بالا گفته شد، هر بازیکن یک قیمت (price) دارد و هر تیمی مقداری پول money دارد و هنگام خرید یک بازیکن باید تیم مورد نظر پول کافی را برای خرید بازیکن داشته باشد و در صورتی که خرید موفقیت آمیز باشد، از پول تیم، به میزان قیمت بازیکن مورد نظر کم می‌شود.

توجه کنید که در امر خرید بازیکن ممکن هست حالت‌های مختلفی پیش بیاید:

- در صورتی که بازیکنی با شناسه‌ی عددی playerID وجود نداشته باشد، باید عبارت player with the id playerID doesnt exist را چاپ کنید.
- در صورتی که تیمی با شناسه‌ی عددی teamID وجود نداشته باشد، باید عبارت team with the id teamID doesnt exist را چاپ کنید.
- در صورتی که تیم مورد نظر، پول کافی برای خرید بازیکن را نداشته باشد باید عبارت the team cant afford to buy this player را چاپ کنید.
- در صورتی که بازیکن مورد نظر، در تیمی عضو باشد، باید عبارت player already has a team را چاپ کنید.

در صورتی که هیچکدام از حالت های بالا پیش نیایند، با موفقیت بازیکن مورد نظر خریداری می‌شود و باید عبارت player added to the team succesfully را چاپ کنید.

**توجه کنید** که ممکن است چندین مورد از حالت‌های بالا پیش بیایند، در این صورت شما باید **اولین** حالتی که پیش می‌آید را در نظر گرفته و خروجی مورد نظر را چاپ کنید و از بقیه حالت‌ها صرف نظر کنید.

## فروش بازیکن

sell playerID teamID

برای فروش بازیکن می‌توان از دستور بالا استفاده کرد. با استفاده از این دستور، تیم با شناسه‌ی عددی teamID بازیکن با شناسه‌ی عددی playerID را به فروش می‌رساند.

همچنین موقع فروش بازیکن ممکن است حالت‌های مختلفی پیش بیایند:

- در صورتی که تیمی با شناسه‌ی عددی teamID وجود نداشته باشد، باید عبارت team doesnt exist را چاپ کنید.
- در صورتی که تیم مورد نظر وجود داشته باشد اما بازیکنی با شناسه‌ی عددی playerID در آن تیم وجود نداشته باشد، باید عبارت team doesnt have this player را چاپ کنید.

در صورتی که حالت‌های بالا پیش نیایند، یعنی با موفقیت بازیکن مورد نظر فروخته شده است و این بازیکن دیگر به تیمی تعلق ندارد (ممکن است در آینده توسط تیمی خریداری شود). در این حالت به اندازه‌ی **قیمت بازیکن** فروخته شده، به پول تیم مورد نظر **افزوده می‌شود**. همچنین در این حالت باید عبارت player sold successfully را چاپ کنید.

## انجام بازی بین دو تیم

match teamID1 teamID2

با استفاده از این دستور، دو تیم با شناسه‌های عددی teamID1 و teamID2 با یکدیگر بازی می‌کنند.

همچنین ممکن است در اجرای این دستور، حالت‌های مختلفی پیش بیاید:

- اگر حداقل یکی از دو تیم وجود نداشته باشد، باید عبارت team doesnt exist را چاپ کنید.
- اگر دو تیم مورد نظر وجود داشتند، اما حداقل یکی از آن‌ها تعداد بازیکن‌هایش **کمتر از ۱۱ نفر** باشد، بازی انجام نمی‌شود و باید عبارت the game can not be held due to loss of the players را چاپ کنید.

در صورتی که هیچکدام از مشکلات بالا پیش نیاید، بازی با موفقیت انجام می‌شود.

در ابتدا برای توصیف روند بازی، نیاز به تعریف پارامتر **قدرت** برای هر تیم داریم.

قدرت تیم اول (تیم با شناسه‌ی عددی teamID1) برابر است با مجموع **سرعت** و **تمام‌کنندگی** ۱۱ بازیکن اول آن تیم.

همچنین قدرت تیم دوم (تیم با شناسه‌ی عددی teamID2) برابر است با مجموع **سرعت** و **دفاع** ۱۱ بازیکن اول آن تیم.

**توجه کنید** که ترتیب بازیکن‌ها در یک تیم، همان ترتیب زمانی اضافه شدن آن‌ها به تیم می‌باشد.

در صورتی که قدرت دو تیم باهم برابر نباشد، تیمی برنده‌ی بازی می‌شود که **قدرت بیشتری** دارد و به تعداد بردهای تیم برنده یک واحد اضافه می‌شود و به تعداد باخت‌های تیم بازنده نیز یک واحد اضافه می‌شود و به تیم برنده، به عنوان جایزه ۱۰۰۰ **واحد** پول داده می‌شود. در صورتی که قدرت دو تیم **برابر باشد**، بازی مساوی می‌شود و تعداد مساوی‌های هر دو تیم یک واحد افزایش پیدا می‌کند.

## نمایش جدول تیم‌ها

rank

با استفاده از دستور بالا، باید جدول رتبه‌بندی تیم‌ها را چاپ کنید.

شما باید تیم‌ها را بر اساس تعداد بردها (هر تیمی که برد بیشتری داشته باشد رتبه بهتری دارد)، و در صورت برابری تعداد بردها، بر اساس تعداد باخت‌های کمتر مرتب کنید (اگر دو تیم بردهای برابری داشته باشند، تیم با باخت کمتر رتبه بهتری می‌گیرد) و به صورت زیر نام آن‌ها را چاپ کنید:

```
1. teamName[0]
2. teamName[1]
...
```

که در آن teamName[0] تیم با رتبه‌ی ۱، teamName[1] تیم با رتبه‌ی ۲ و... می‌باشند.

توجه کنید که تعداد مساوی‌ها مهم نمی‌باشد و همچنین اگر دو تیم تعداد باخت‌ها و بردهایشان برابر بود، تیمی که شناسه‌ی عددی کمتری دارد رتبه‌ی بهتری می‌گیرد.

## پایان لیگ

end

در انتها، پس از تمام شدن لیگ، عبارت بالا در خروجی داده می‌شود و به این معنی است که ورودی‌ها تمام شده است و برنامه شما نیز پایان می‌یابد.

## ورودی

ورودی شامل چندین خط می‌باشد که در هر خط یکی از دستوراتی که در صورت سوال گفته شد، داده می‌شود.

همچنین در خط آخر ورودی، دستور end داده می‌شود.

توجه کنید که ممکن است در ورودی داده شده، بعضی از خط‌ها خالی باشند و هیچ دستوری در آن‌ها نباشد.

## خروجی

پس از اجرای برخی از دستورها، باید خروجی‌هایی را چاپ کنید که در توضیحات هر دستور در صورت نیاز گفته شده است. توجه کنید که هر خروجی باید در یک خط چاپ شود و از چاپ تمامی خروجی‌ها در یک خط پرهیز کنید.

## مثال

### ورودی نمونه ۱

```
new player ArshiA 1 2 3 4
new player sadegh 1 2 3 4
new player haniye 1 2 3 4
new player fateme 1 2 3 4
new player qorban 1 2 3 4
new player abbasi 1 2 3 4
new player alikmr 1 2 3 4
new player sadegh 1 2 3 4
new player hamide 1 2 3 4
new player karane 1 2 3 4
new player vanaki 1 2 3 4

new player Ars 1 100 100 100
new player sad 1 100 100 100
```

```
new player han 1 100 100 100
new player fat 1 100 100 100
new player qor 1 100 100 100
new player abb 1 100 100 100
new player ali 1 100 100 100
new player sad 1 100 100 100
new player ham 1 100 100 100
new player kar 1 100 100 100
new player van 1 100 100 100
```

```
new team t2 100
new team t1 100
buy 12 1
buy 13 1
buy 14 1
buy 15 1
buy 16 1
buy 17 1
buy 18 1
buy 19 1
buy 20 1
buy 21 1
buy 22 1
```

```
buy 1 2
buy 2 2
buy 3 2
buy 4 2
buy 5 2
buy 6 2
buy 7 2
buy 8 2
buy 9 2
buy 10 2
buy 11 2
match 1 2
rank
end
```

خروجی نمونه ۱

[illegible]

1. t2

2. t1



# دل جنگی

.