

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 `i`-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. t_2
2. t_1

اما چیجوری؟

• محدودیت زمان: ۱ ثانیه

• محدودیت حافظه: ۲۵۶ مگابایت

عبارات در حالت عادی به صورت میانوند نوشته می شوند. یعنی هر عملگر (غیر از عملگر یکتایی) بین عملوندهای خود قرار می گیرد. یکی از مشکلات محاسبه اینگونه عبارات برای کامپایلر ها وجود پرانتز های تو در تو است . به این دلیل در بعضی کامپایلرها برای سهولت ترجمه، نخست این عبارات به شکل پسوند تبدیل می شوند. در عبارات پسوند عملگر بعد از دو عملوند خود قرار دارد. البته حالت پیشوند هم وجود دارد که در این حالت عملگر قبل از عملوندهایش می باشد. به مثال زیر توجه کنید:

• عبارت میانوند: $(A+B)*(C-D)$

• عبارت پیشوندی: $*+AB-CD$

• عبارت پسوندی: $AB+CD-*$

برنامه ای بنویسید که ابتدا رشته ای از کاربر بگیرد این رشته باید پیشوندی (prefix) یا پسوندی (postfix) باشد. در ادامه رشته عبارت را کاربر وارد میکند. کار برنامه محاسبه عبارت بر مبنای رشته اول به دو فرم گفته شده است. در نهایت مقدار محاسبه شده عبارت برای کاربر چاپ میشود.

*** نکته مهم :** برای محاسبه عبارت تنها از آرایه ها استفاده کنید. *

***توجه:** برای این سوال مجاز به استفاده از آرایه با طول ثابت نیستید و مدیریت حافظه باید بصورت پویا انجام شود. همچنین زمانی که محاسبات انجام میشوند و خانه ای از آرایه خالی میشود باید از متد free استفاده کنید و در هر مرحله طول آرایه را کاهش دهید. *

*** نکته مهم :** تضمین میشود تمام اعداد صحیح باشند. *

ورودی

در ورودی در خط اول یک رشته میگیریم که نشانگر نوع عبارت محاسباتی است و در خط بعدی عبارتی که باید

حاصل آن محاسبه شود را دریافت میکنیم.

خروجی

در خروجی حاصل عبارت گفته شده نمایش داده میشود.

مثال

ورودی نمونه ۱

postfix

10 2 ^ 12 12 * + 0 -

خروجی نمونه ۱

244

ورودی نمونه 2

prefix

- + / 7 1 5 ^ 4 ^ 2 3

خروجی نمونه 2

-65524