

حاج زنبور عسل

• محدودیت زمان: ۱ ثانیه

• محدودیت حافظه: ۲۵۶ مگابایت

حاج زنبور عسل که مدت‌ها به دنبال مادرش بود، بالاخره از پیدا کردن او ناامید شد و حالا تصمیم به تشکیل خانواده گرفته است. به او کمک کنید که کندویی برای خانواده‌اش درست کند. اما ساختن کندو ساده نیست! او باید کندویی با **شکل دقیق و منظم** بسازد تا زنبورهای دیگر بتوانند به راحتی در آن زندگی کنند. حاج زنبور عسل به کمک شما نیاز دارد تا کندوی خود را طراحی کند.

یک عدد صحیح مثبت n به شما داده می‌شود که نشان‌دهنده تعداد "لایه‌های" کندوی حاج زنبور عسل است. وظیفه شما این است که کندوی او را طوری طراحی کنید که این ویژگی‌ها را داشته باشد:

۱. کندو باید به صورت متقارن از بالا به پایین باشد.

۲. هر لایه باید با توجه به موقعیت خود تعداد مشخصی از "خانه‌های شش‌ضلعی" (نمایش داده شده با *) داشته باشد.

ورودی: یک عدد صحیح n ($1 \leq n \leq 10$) که تعداد لایه‌های کندو را مشخص می‌کند.

خروجی: یک الگوی کندوی شش‌ضلعی متقارن که حاج زنبور عسل بتواند از آن برای ساخت خانواده استفاده کند.

مثال

ورودی نمونه ۱

1

خروجی نمونه ۱

*

ورودی نمونه ۲

3

خروجی نمونه ۲

```
* * *
* * * *
* * * * *
* * * *
* * *
```

ورودی نمونه 3

6

خروجی نمونه 3

```
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
```

سینما سازمانی

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

کارکنان کوئرا می‌خواهند به یک سینما که گنجایش k نفر را دارد، بروند. مشکل این است که برای هر کارمند کوئرا که به سینما برود، او باید حتماً به همراه تمام دوستانش برود. اگر کارمند i ام کوئرا a_i دوست داشته باشد (به جز خودش)، حداکثر چند نفر از اعضای کوئرا می‌توانند همزمان به سینما بروند؟



ورودی

در سطر اول ورودی، دو عدد طبیعی n و k که به ترتیب تعداد کارکنان کوئرا و گنجایش سینما را نشان می‌دهند، داده می‌شود.

$$1 \leq n \leq 100\,000$$

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

در سطر بعدی n عدد صحیح داده می‌شود که عدد i ام تعداد دوستان کارمند i ام کوئرا است.

$$0 \leq a_i \leq 10^9$$

خروجی

در تنها سطر خروجی حداکثر تعداد کارمندان کوئرا که می‌توانند با هم به سینما بروند را چاپ کنید.

مثال‌ها

ورودی نمونه ۱

```
5 10
4 3 0 2 1
```

خروجی نمونه ۱

```
4
```

اگر همه‌ی ۵ کارمند کوئرا با دوستانشان بیایند، ۱۵ نفر می‌شوند ولی ظرفیت سینما ۱۰ نفر است. اما اگر کارمند ۱ و دوستانش نیایند دقیقاً ۱۰ نفر می‌شوند. بنابراین پاسخ مسئله ۴ یعنی تعداد کارمندانی که به سینما می‌آیند، است.

لیوان بازی

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

برنامه نویسان رهنما در اوقات فراغت و استراحتشان بازی های زیادی برای سرگرمی انجام می دهند. یکی از این بازی ها *لیوان بازی* است. لیوان بازی یک بازی دونفره است به این صورت که در ابتدا سه لیوان چینی داریم که در یک ردیف به صورت برعکس قرار گرفته اند و یک عدد نخود زیر یکی از آن هاست. ابتدا نفر اول به نفر دوم اعلام میکند که نخود زیر کدام لیوان است. سپس طی یک سری حرکت ، هر مرحله جای یک لیوان را با لیوان دیگر عوض میکند و وقتی حرکاتش تمام شد نفر دوم باید بگوید که نخود زیر کدام لیوان است. بدیتهای لیوان چینی شفاف نیست و نفر دوم نمیتواند ببیند که نخود زیر کدام لیوان است. حال ما از شما میخواهیم به نفر دوم کمک کنید تا بتواند بگوید که پس از انجام حرکات نخود زیر کدام لیوان است.

ورودی

ابتدا در یک خط x, n را به شما میدهیم که n تعداد حرکات نفر اول است و x که یکی از کاراکترهای L, M, R است که نشان میدهد در ابتدا نخود زیر لیوان چپی ، وسطی یا راستی است. سپس در n خط ، که هر خط نشان دهنده یک حرکت است ، در هر خط دو کاراکتر متفاوت به شما داده میشود که نشان میدهد که نفر اول در آن حرکت کدام لیوان ها را با هم عوض میکند. کاراکتر L نشان دهنده لیوان چپی است. کاراکتر M نشان دهنده لیوان وسطی است. کاراکتر R نشان دهنده لیوان راستی است.

تضمین میشود که تمام کاراکتر های موجود در ورودی یکی از مقادیر L, M, R را دارند و همچنین:

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

خروجی

در یک خط یک کاراکتر چاپ کنید که نشان دهد در پایان حرکات ، نخود زیر کدام لیوان است. اگر در پایان نخود

زیر لیوان چپ بود شما باید L چاپ کنید. اگر در پایان نخود زیر لیوان وسط بود شما باید M چاپ کنید. اگر در پایان نخود زیر لیوان راست بود شما باید R چاپ کنید.

مثال

ورودی نمونه ۱

3 M
L M
R L
M L

خروجی نمونه ۱

R

توضیح : ابتدا نخود زیر لیوان وسطی قرار دارد. با انجام اولین حرکت جای لیوان وسطی و چپی عوض میشود پس در پایان حرکت اول نخود زیر لیوان چپ قرار میگیرد. با انجام دومین حرکت جای لیوان راستی و چپی عوض میشود پس در پایان حرکت دوم نخود زیر لیوان راست قرار میگیرد. با انجام سومین حرکت جای لیوان چپی و وسطی عوض میشود و از آنجایی که نخود زیر لیوان راستی بود جایش تغییر نمیکند و در پایان نخود زیر لیوان راستی قرار میگیرد.

ورودی نمونه ۲

5 L
L M
L M
R M
R L
R M

خروجی نمونه ۲

M

حلقه

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

فرض کنید یک **لینک لیست تک مسیره** دارید که از گره‌های مختلف تشکیل شده است. هر گره شامل یک مقدار (value) و یک اشاره‌گر به گره بعدی است. لینک لیست ممکن است به یکی از گره‌های قبلی خود اشاره کند و یک حلقه تشکیل دهد.

سوال:

برنامه‌ای بنویسید که تشخیص دهد آیا لینک لیست دارای حلقه است یا خیر. برای حل این مسئله، اجازه‌ی استفاده از هیچ ساختار داده اضافی (مانند آرایه، مجموعه، نقشه و غیره) یا حافظه اضافی برای نگهداری گره‌های بازدید شده را ندارید. تنها باید از اشاره‌گرها و متغیرهای ساده استفاده کنید.

نکات:

- شما فقط به **سر لینک لیست** (head) دسترسی دارید. (طبیعتاً از آن می‌توانید به نودهای بعدی بروید.)
- لینک لیست می‌تواند بدون حلقه باشد یا دارای یک حلقه در هر نقطه از لیست.
- در این سوال از شما کد خواسته نشده بلکه فقط باید راه حل خود را به صورت pdf ارسال کنید.

مثال

ورودی نمونه ۱

1 -> 2 -> 3 -> 4 -> 5 -> 3

خروجی نمونه ۱

loop

ورودی نمونه ۲

1 -> 2 -> 3 -> 4 -> 5

خروجی نمونه ۲

no loop

جایزه

• محدودیت زمان: ۱ ثانیه

• محدودیت حافظه: ۱۲۸ مگابایت

زانا سالی یک بار یک جشن خاص برگزار می‌کند و تعدادی از دوستانش را به این جشن دعوت می‌کند. اسم این جشن «جشن هدیه‌ها» است! هر فردی که در این جشن شرکت می‌کند مقداری پول به همراه خود دارد و به تعدادی از دوستانش هدیه می‌دهد. روش هدیه دادن در این جشن کمی عجیب است! هر کدام از افراد یک لیست هدیه دارد که در آن لیست، نام تعدادی از دوستانش که در جشن شرکت کرده‌اند نوشته شده است و تمام پولی که همراه دارد را بین افراد این لیست به طور مساوی تقسیم می‌کند و این پول را به آنها هدیه می‌دهد! چون پول اعشاری (کوچکتر از یک) نداریم، این تقسیم‌ها تقسیم صحیح هستند و اگر تقسیم پول بین اعضای لیست باقیمانده‌ای داشته باشد، فرد هدیه دهنده این باقیمانده را برای خود نگه می‌دارد. به طور مثال اگر ساینا ۱۱ واحد پول داشته باشد و در لیست او فقط سه نفر باشند، به هر کدام از آنها ۳ واحد پول می‌دهد و ۲ واحد از پول خود را برای خود نگه می‌دارد.

حال شما برنامه‌ای بنویسید که پس از گرفتن اسامی شرکت کنندگان، مقدار پول اولیه‌ی هر کدام و لیست هدیه هر کس، مشخص کند که هرکسی چقدر سود یا زیان کرده است!

ورودی

- خط 1: عدد n که برابر است با تعداد شرکت کنندگان در جشن.
- خط 2 تا $n+1$: در هر خط اسم یکی از شرکت کنندگان.
- خط $n+1$ الی آخر: از این خط به بعد ورودی به n دسته تقسیم می‌شود که هرکدام مطابق زیر است: خط اول نام فردی که قرار است هدیه بدهد. در خط دوم دو عدد می‌آید: عدد اول مقدار پول آن فرد، عدد دوم (k) تعداد افراد موجود در لیست هدیه‌ی آن فرد در k خط بعدی در هر خط نام یکی از افراد موجود در لیست هدیه‌ی آن فرد.

می‌توانید فرض کنید نام هر دو نفر از افراد شرکت‌کننده در جشن متمایز است و

$$2 \leq n \leq 10$$

خروجی

در خروجی باید n خط چاپ کنید که در هر ابتدای هر خط نام هر شخص و بعد از آن مقدار سود او آورده شود.
(اگر آن شخص ضرر کرده است، باید منفی مقدار ضرر چاپ شود.) ترتیب نام‌ها در خروجی باید مانند ترتیب نام‌ها در خطوط 2 تا $n+1$ ورودی باشد.

مثال

ورودی نمونه

```
5
dave
laura
owen
vick
amr
dave
200 3
laura
owen
vick
owen
500 1
dave
amr
150 2
vick
owen
laura
0 2
amr
vick
vick
```

0 0

خروجی نمونه

dave 302
laura 66
owen -359
vick 141
amr -150

فروشگاه کتاب

- محدودیت زمان: ۱ ثانیه

- محدودیت حافظه: ۲۵۶ مگابایت

فرض کنید یک ردیف کتاب داریم که آنها را در یک قفسه از کتابخانه از چپ به راست چیده‌ایم. در این کتابخانه برای گذاشتن و برداشتن کتاب‌ها نظم خاصی وجود دارد به این صورت که اگر بخواهیم یک کتابی را در این قفسه بگذاریم فقط می‌توانیم آن را سمت چپ یا سمت راست کتاب‌ها بگذاریم و برای برداشتن نیز فقط می‌توانیم از سمت چپ کتاب برداریم.

در اصل می‌توانیم سه عمل زیر را روی این ردیف کتاب انجام دهیم:

۱. عبارت $AddRight\ X$: در این عمل کتاب با نام X را به سمت راست کتاب‌ها اضافه می‌کنیم.

۲. عبارت $AddLeft\ X$: در این عمل کتاب با نام X را به سمت چپ کتاب‌ها اضافه می‌کنیم.

۳. عبارت $RemoveLeft$: در این عمل کتاب سمت چپ را از قفسه برمی‌داریم.

می‌خواهیم برنامه‌ای بنویسیم که ابتدا یک دسته کتاب را به عنوان ورودی گرفته و سپس تعدادی از عمل‌های بالا را روی آن انجام دهد و دسته کتاب نهایی را به عنوان خروجی چاپ کند.

برای پیاده‌سازی این برنامه لازم است از داده‌ساختار لیست پیوندی استفاده کنید.

لیست پیوندی (Linked list) ساختاری شامل دنبال‌های از عناصر است که هر عنصر دارای اشاره‌گری به عنصر بعدی در دنباله است.

برای پیاده‌سازی لیست پیوندی برای این مسئله یک `struct Book` تعریف می‌کنید که شامل یک `string` `Name` برای نگهداری نام کتاب و یک `Book* Next` برای اشاره به عنصر بعدی است.

سمت چپ‌ترین کتاب را اولین کتاب و راست‌ترین کتاب را آخرین کتاب در نظر می‌گیریم و همچنین کتاب بعدی هر کتاب را کتاب بلافاصله سمت راست آن در نظر می‌گیریم. دو اشاره‌گر مانند `Book* first` و `Book* last` برای اشاره به کتاب اول و کتاب آخر نگهداری می‌کنیم. پس از هر عمل حذف یا اضافه در سمت چپ یا راست یکی

از این دو اشاره‌گر باید به روزرسانی شوند.

```
struct Book
{
    string Name;
    Book *Next;
};
```

برای هر عمل اضافه کردن کتاب ، باید از دستور `new` برای گرفتن حافظه برای `Book` جدید استفاده کنید و برای هر عمل حذف ، برای جلوگیری از نشت حافظه باید کتاب مورد نظر را با استفاده از دستور `delete` از حافظه پاک کنید.

در ادامه لیست دستوراتی که به برنامه داده می‌شود و مفهوم آنها آمده است:

command	description
AddLeft BookName	با دیدن این عبارت، باید یک کتاب به ابتدای کتابخانه (سمت چپ) اضافه شود
AddRight BookName	با دیدن این عبارت، باید یک کتاب به انتهای کتابخانه (سمت راست) اضافه شود
DeleteLeft	با دیدن این عبارت، باید چپترین کتاب در کتابخانه را حذف کنید
Exit	با دیدن این کاراکتر، برنامه به پایان میرسد و ابتدا تعداد کتابهای داخل کتابخانه را چاپ کنید و سپس لیست کتابهای داخل کتابخانه را به ترتیب از چپ به راست چاپ کنید

ورودی

ابتدا یک عدد n در ورودی داده می‌شود که نشانگر تعداد کتابهای داخل کتابخانه در ابتدای کار است سپس n رشته به ترتیب چپ به راست که هر کدام نام یکی از کتابهاست. (نام هر کتاب رشته‌ای به طول حداکثر ۱۳۳ می‌باشد و از حروف کوچک و بزرگ انگلیسی و اعداد تشکیل شده است). (ممکن است در نام یک کتاب *space* نیز وجود داشته باشد). سپس در هر مرحله یکی از دستورات بالا داده می‌شود.

تعداد دستوراتی که به برنامه داده می‌شود حداکثر 10^6 می‌باشد.

خروجی

در سطر اول تعداد کتاب‌های موجود در کتابخانه چاپ شود. در سطرهای بعدی در هر سطر نام یک کتاب از کتاب‌های موجود در کتابخانه چاپ شود. ترتیب چاپ کتاب‌ها از چپ به راست می‌باشد.

مثال

ورودی نمونه

```
3
Mathematics
General Physics 2
Advanced Programming
DeleteLeft
AddLeft
Kelile va Demne
AddRight
Boostane Hafez
Exit
```

خروجی نمونه

```
4
Kelile va Demne
General Physics 2
Advanced Programming
Boostane Hafez
```

linklistestan

- محدودیت زمان: ۱ ثانیه

- محدودیت حافظه: ۲۵۶ مگابایت

در دنیای موازی‌ای که در آن برنامه‌نویسی حکم یک زبان جادویی را دارد، **پروژه‌های نرم‌افزاری** به‌طور مستقل در سیاره‌ای به نام "**دنیای لینک‌لیست**" زندگی می‌کنند. این پروژه‌ها باید برای حل مشکلات پیچیده، به سرعت به دستوراتی که به آنها داده می‌شود، پاسخ دهند. یکی از این پروژه‌ها به نام "**کارگزار داده‌ها**" وظیفه‌ای بسیار چالش‌برانگیز پیدا کرده است: **باید یک سری داده را در لینک‌لیستی پیچیده مدیریت کند** و دستوراتی برای تغییر و اصلاح این لیست را اجرا کند.

پروژه کارگزار داده‌ها، در ابتدا با لیستی از اعداد مواجه می‌شود که در یک لینک‌لیست قرار دارند مثل : 2 3 4 6 7 اما او نمی‌داند که به زودی، **دستورات عجیب و غریبی** از طرف مدیر پروژه به او داده می‌شود که نه تنها باید به‌طور دقیق اجرا کند، بلکه باید درک کند که هر دستور چه تاثیری روی ساختار این لیست خواهد داشت. به همین دلیل، کارگزار داده‌ها باید **مهارت‌های خود در مدیریت لینک‌لیست‌ها** را افزایش دهد.

دستورات جدید:

دستور اول: مدیر پروژه به کارگزار داده‌ها می‌گوید:

2 end

یعنی باید عدد 2 را به انتهای لیست اضافه کند. کارگزار داده‌ها باید این دستور را با دقت اجرا کند و در انتهای لینک‌لیست قرار دهد

دستور دوم: بعد از گذشت چند لحظه، مدیر پروژه دستور می‌دهد:

4 start

بدین معنا که کارگزار داده‌ها باید عدد 4 را در ابتدای لیست قرار دهد. این دستور برای تغییر ترتیب داده‌ها و ترتیب‌دادن مجدد آنها طراحی شده است.

دستور سوم :

3 before 2

در این دستور، کارگزار داده‌ها باید **عدد 3 را قبل از اولین 2** که در لیست قرار دارد، وارد کند. به این معنا که:

• اگر لیست اولیه به شکل 2 3 7 6 4 باشد و بعد از اجرای این دستور، باید لیست به شکل

• 2 3 3 7 6 4 تغییر کند

دستور چهارم: دستور دیگری از طرف مدیر پروژه وارد می‌شود:

4 after 3

#این سوال حتما باید توسط لینک‌لیست حل شود. #استفاده از توابع آماده مجاز نیست و خودتان باید لینک لیست و توابع گفته شده را پیاده سازی کنید.

ورودی

در این بخش قالب ورودی و محدودیت‌های آن توضیح داده شود.

شروع ورودی شامل یک خط است که در آن دو عدد طبیعی n و m با فاصله از هم آمده است. که n نشان دهنده‌ی تعداد اعداد لیست است و m تعداد دستوراتی که داده خواهد شد. در خط بعد n تا عدد و در m خط بعدی دستورات وارد خواهند شد.

$$1 \leq n, m \leq 100$$

خروجی

خروجی برنامه‌ی شما باید شامل 1 خط باشد که در آن اعداد لینک لیست را از اول تا آخر مینویسد.

مثال**ورودی نمونه ۱**

3 4
1 4 5
4 after 4
4 before 4
4 end
2 start

خروجی نمونه ۱

2 3 4 4 4 4

ورودی نمونه ۲

5 3
1 2 6 4 5
4 after 4
5 start
2 start

خروجی نمونه ۲

2 5 1 2 6 4 4 5