

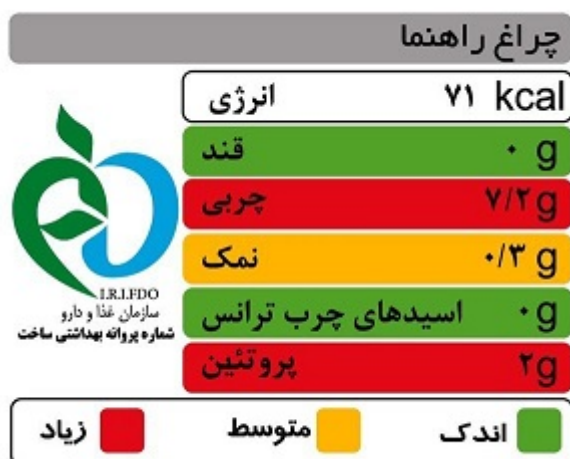
رژیم سخت

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

لیته، که مدت‌ها پیش دلش را به **فیته** باخته بود، به تازگی متوجه شده‌است که فیته به خاطر اضافه‌وزن بیش از حد لیته به او اهمیتی نمی‌دهد. بنابراین لیته تصمیم گرفته‌است که در اسرع وقت وزن و هیکل خودش را به ایده‌آل فیته برساند.

پس از مشاوره‌های فراوان، لیته به این نتیجه می‌رسد که به هیچ وجه نباید خوراکی‌هایی که برچسب راهنمای سلامتشان خطرناک است را بخورد. برچسب راهنمای سلامت به این صورت است که اطلاعاتی در مورد قند، چربی، نمک، اسیدهای چرب ترانس و پروتئین می‌دهد. و می‌دانیم که یک برچسب سلامت خطرناک است اگر حداقل یکی از شرایط زیر برقرار باشد:

- حداقل سه مورد قرمز باشند.
- حداقل دو مورد قرمز و حداقل دو مورد زرد باشند.
- همه موارد زرد یا قرمز باشند.



لیته که از بچگی یکی از خوره های تکنولوژی بود، می خواهد برنامه‌ای برای ساعت هوشمندش بنویسد که موقع خرید این خوراکی‌ها به او هشدار بدهد. اما چون این روزها فکرش خیلی درگیر فیته است تمرکز ندارد و از شما می‌خواهد در نوشتن این برنامه به او کمک کنید.

ورودی

ورودی تنها شامل یک سطر است که در آن برچسب سلامت به صورت یک رشته متشکل از پنج حرف آمده است؛ R نشان دهنده‌ی رنگ قرمز، Y نشان دهنده‌ی رنگ زرد، و G نشان دهنده‌ی رنگ سبز است.

خروجی

در صورتی که برچسب ورودی یک برچسب خطرناک باشد در تنها سطر خروجی عبارت `nakhor lite` را چاپ کنید و در غیر این صورت عبارت `rahat baash` را چاپ کنید.

ورودی نمونه ۱

GGGGG

خروجی نمونه ۱

`rahat baash`

در نمونه‌ی بالا، همه‌ی موارد سبز هستند و خوردن این خوراکی هیچ خطری ندارد.

ورودی نمونه ۲

RYRYR

خروجی نمونه ۲

`nakhor lite`

خوراکی بالا هر سه شرط گفته شده را دارد که حتی با داشتن یکی از آن‌ها خطرناک می‌شد؛ پس خیلی خطرناک است!

پرگوشت

تنبوله از بچگی عاشق خوردن گوشت به مقدار غیرلازم بوده و هنوز هم دست از این عادت اشتباهش برنداشته. به هر حال او رئیس قبیله‌ی «گاوچران‌های بی‌ادعا»ست و قدرت این را دارد که هرچقدر که دوست دارد، گوشت بخورد! نوش جان‌ش اصلاً، به ما چه (:)



روزی در یک مهمانی خانوادگی، دو ظرف قورمه‌سبزی با تعداد متفاوتی گوشت در هر کدام، سر سفره آورده می‌شود. تنبوله نمی‌داند کدام ظرف تعداد گوشت بیشتری دارد. او می‌خواهد در سمتی از سفره بنشیند که ظرف با تعداد گوشت بیشتر وجود دارد.

تنبوله ۱۷۳ سال عمر کرده و چشمانش ضعیف شده، برای همین فرق بین لیموامانی و گوشت را به سختی متوجه می‌شود. اگر او به شما شکل و شمایل ظرف‌های قورمه‌سبزی را بدهد، آیا شما می‌توانید تعداد گوشت‌های هر ظرف قورمه‌سبزی را به او بگویید؟ فقط لطفاً سریع‌تر، تنبوله بسیار گشنه‌ست و طاقت دوری از قورمه‌سبزی را ندارد.

هر ظرف قورمه‌سبزی به شکل یک مستطیل $m \times n$ است که در هر خانه از آن یکی از کاراکترهای * یا o یا آمده. کاراکتر * به معنای گوشت، کاراکتر o به معنای لیموامانی و کاراکتر . به معنای سبزی است.

ورودی

در یک سطر دو عدد صحیح n و m داده می‌شود. در n سطر بعدی ظرف اول قورمه‌سبزی داده می‌شود که در هر سطر m کاراکتر بدون فاصله می‌آید. که این کاراکترها $*$ یا o و یا $.$ هستند.

پس از آن دوباره n سطر می‌آید که نشان‌دهنده‌ی ظرف دوم قورمه‌سبزی است و در هر سطر m کاراکتر بدون فاصله می‌آید.

خروجی

در تنها سطر خروجی، به ترتیب تعداد گوشت‌های ظرف اول و تعداد گوشت‌های ظرف دوم را چاپ کنید.

مثال

ورودی نمونه یک

```
3 4
*.*
*.oo
o*.o
*.*o
o.*o
.*.o
```

خروجی نمونه یک

```
5 4
```

ورودی نمونه دو

```
5 1
.
o
.
o
*
```

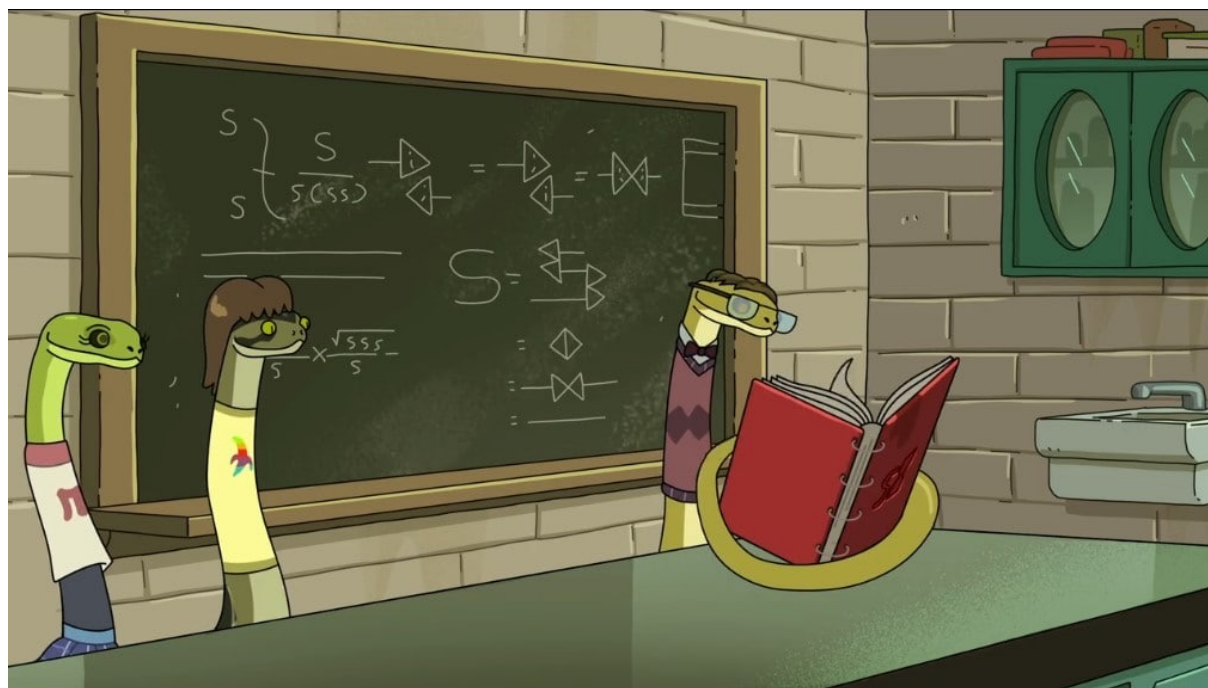
*
*
*
*
O

خروجی نمونه ۲

1 4

سیاره کم زبان ها

- محدودیت زمان: ۵.۵ ثانیه
- محدودیت حافظه: ۵ مگابایت



علی کار درست برای تست سفینه ی فضایی که ساخته راهی سفری فضایی شده همه جی خوب پیش می رود تا اینکه به صورت ناگهانی آلام وضعیت قرمز سفینه روشن می شود!! بنابراین علی نیاز دارد تا برای نجات جان خودش بر روی سیاره ی کم زبان ها فرود بیاید! ساکنین این سیاره - کمزبان ها - دارای زبانی بدوی هستند. کل این زبان از 9 کاراکتر تشکیل شده است و حتی فاصله نیز در جملات زبان آنها جایگاهی ندارد. ریک پس از مدتی کلنچار رفتن با کمزبان ها، تصمیم می گیرد نرم افزار تولید کند که به او در فهم زبان کمزبان ها یاری رساند و جایگاه واژگانی که ریک از قبل می شناسد را در جمله پیدا کند تا بلکه او بتواند جملات آنها را متوجه شود. علی کار درست برای ساخت این نرم افزار به کمک شما احتیاج دارد.

این نرم افزار با دریافت یک جمله و یک کلمه، محل قرارگیری کلمه در جمله را پیدا می کند و اگر کلمه در جمله یافت نشود عدد 1- را چاپ می کند.

ورودی

ابتدا در یک خط عدد n داده می‌شود که نشان دهنده طول جمله‌ای است که قرار است دریافت کند. جمله ورودی فقط متشکل از کاراکتر های $a, b, c, d, e, f, g, h, i$ است و شامل فاصله و یا کاراکترهای دیگر نمی‌باشد. سپس در خط دوم ورودی، عدد m داده می‌شود که نشان‌دهنده طول کلمه است. قوانین گفته شده درباره جمله درباره کلمه نیز صدق می‌کند.

$$0 < m, n < 10$$

سپس ابتدا در n خط، کاراکتر های جمله به ترتیب داده می‌شوند. (در هر خط یک کاراکتر) و بعد از آن در m خط، کاراکتر های کلمه به ترتیب داده می‌شوند. (در هر خط یک کاراکتر) همچنین تضمین می‌گردد کلمه حداکثر یک بار در جمله ظاهر می‌گردد.

خروجی

در تنها خط خروجی باید یک عدد بین 0 تا $n - m$ داده شود که نشان دهنده این است که کلمه در چندمین اندیس جمله آمده است. اگر کلمه در جمله یافت نمی‌شد، باید عدد -1 چاپ گردد. منظور از اینکه کلمه در چندمین اندیس جمله ظاهر شده است این است که چنانچه کلمه در جمله آمده‌است، جایی که کلمه شروع می‌شود چندمین حرف از جمله است (و شمردن اندیس از 0 آغاز می‌شود نه 1) (مثلا در **test** ، اگر دنبال **es** باشیم می‌گوییم از اندیس اول شروع شده، **te** از اندیس صفر شروع شده و ...)

مثال

ورودی نمونه ۱

5
2
a
b
c
d
e
c
d

خروجی نمونه ۱

2

در این مثال جمله داده شده **abcde** است و کلمه **cd** است. کلمه ما در جمله ظاهر شده و اندیس دوم جمله است.

ورودی نمونه ۲

9

3

a

a

a

b

b

b

i

i

i

b

b

i

خروجی نمونه ۲

4

جمله داده شده **aaabbbiii** است و کلمه داده شده **bbi** است. کلمه در جمله ظاهر شده و در اندیس چهارم قرار گرفته است.

ورودی نمونه ۳

5

3

a

a

e
f
h
a
e
g

خروجی نمونه ۳

-1

جمله داده شده **aaefh** است و کلمه داده شده **aeg** است. این کلمه اصلا در جمله ظاهر نشده است پس خروجی 1- است.

دور باستان

در آخرین اکتشافات باستان‌شناسان در سرزمین «کهن»، یک دفترچه بسیار مرموز پیدا شده است. طی تحقیقات بسیار مشخص شد که این دفترچه، دفترچه خاطرات رستم است. پس از ناکام ماندن تلاش باستان‌شناسان برای خواندن خاطرات رستم، مشخص شد که رستم از بیم خوانده شدن خاطراتش، متن این دفترچه را با الگوریتم‌های بسیار پیچیده رمزنگاری کرده است و کلید این رمز را در یکی از اهرام «دور» قرار داده است.

این کلید امروزه تحت تدابیر شدید امنیتی در موزه «دور باستان» نگهداری می‌شود و این موزه به هیچ وجه حاضر نیست این کلید را در اختیار باستان‌شناسان سرزمین «کهن» قرار دهد.

پس از اصرارهای زیاد باستان‌شناسان «کهن»، موزه «دور باستان» حاضر شد این کلید را در اختیار آن‌ها بگذارد. بنابراین درب سالن محل نگه‌داری این کلید را باز کرد تا بروند و این کلید را بردارند. اما باستان‌شناسان باهوش «کهن» متوجه شدند در مسیر درب سالن تا خود کلید تعدادی تله قرار داده شده است. آنها با تجهیزات پیشرفته‌ی خود، مکان این تله‌ها را پیدا کرده‌اند و اکنون می‌خواهند همه مسیرهای ممکن برای رسیدن به کلید را بررسی کنند و بهترین مسیر را انتخاب کنند.

B			T	T				T	T	T	T				K
---	--	--	---	---	--	--	--	---	---	---	---	--	--	--	---

این سالن به شکل یک جدول $1 \times n$ است که در خانه‌ی شماره ۱، باستان‌شناسان «کهن» قرار دارند (با نماد B) و در خانه‌ی شماره n ، کلید قرار دارد (با نماد K) و در برخی از خانه‌های دیگر تله وجود دارد (با نماد T). اگر باستان‌شناسان وارد خانه‌ی تله‌دار بشوند، به پایین می‌افتند و خوراک کوسه‌ها می‌شوند.

باستان‌شناسان «کهن» در هر حرکت می‌توانند یکی از این سه کار را انجام دهند:

- یک خانه به جلو بروند.
- از خانه جلویی بپرند (مستقیماً به دو خانه جلوتر بروند).
- از دو خانه جلویی بپرند (مستقیماً به سه خانه جلوتر بروند).

برنامه شما باید تعداد همه راه‌های ممکن برای رسیدن باستان‌شناسان به کلید را محاسبه کند.

ورودی

در خط اول، عدد n می‌آید و در خط بعدی نقشه‌ی سالن به شکل یک رشته به طول n از حروف B و T و K و . می‌آید. حرف . نماد خانه‌های خالی است. مقدار n حداکثر ۲۰۰ خواهد بود.

خروجی

در خروجی تعداد راه‌های ممکن برای رسیدن باستان‌شناسان به کلید را بنویسید.

ورودی نمونه ۱

21

B..T.T....TT.T.T....K

خروجی نمونه ۱

198

ورودی نمونه ۲

5

B..TK

خروجی نمونه ۲

3

ورودی نمونه ۳

30

B..T....TT..T.T...TTT..T.T..K

خروجی نمونه ۳

0

ورودی نمونه ۴

50

B..T..T.T..TT...T.T..T.TT.T....T.T...TT.T..T....TK

خروجی نمونه ۴

93312

ب.ب.م.م

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

مصطفی که آدمی کاری است، میتواند بزرگترین مقسوم علیه مشترک دو عدد طبیعی را به راحتی حساب کند. او که به تازگی n عدد طبیعی دریافت کرده است، میخواهد برای هنرنمایی دو تا از آنها را انتخاب کند و بزرگترین مقسوم علیه مشترک آنها را اعلام کند. میدانیم مصطفی طوری این دو عدد را انتخاب میکند که عدد اعلامی اش بیشترین مقدار ممکن را داشته باشد. شما باید با گرفتن اعداد دریافتی مصطفی، عددی که اعلام میکند را پیش بینی کنید.

ورودی

در سطر اول ورودی عدد n می‌آید که نشان دهنده تعداد اعداد است. سپس در سطر بعدی n عدد طبیعی x_1, x_2, \dots, x_n می‌آیند که اعداد دریافتی مصطفی هستند.

$$2 \leq n \leq 100\,000$$

$$1 \leq x_i \leq 100\,000$$

خروجی

در تنها سطر خروجی باید عدد اعلامی مصطفی چاپ شود.

مثال

ورودی نمونه

5

1 2 3 4 5

خروجی نمونه

2

بزرگترین مقسوم علیه مشترک ۲ و ۴ برابر ۲ است، همچنین بزرگترین مقسوم علیه مشترک سایر جفت ها برابر ۱ است.

پشنج پلاس

- محدودیت زمان: ۰.۵ ثانیه
- محدودیت حافظه: ۵۰ مگابایت

پشنج می‌خواهد در هزینه‌هایش صرفه‌جویی کند؛ او برای این کار، مسیرهایی که در طول روز از آن‌ها عبور می‌کند را در نظر می‌گیرد و سعی می‌کند آن‌ها را مرتب کند.

او محل‌هایی که در آن‌ها رفت و آمد دارد را با توجه به مسافت آن محل تا خانه‌اش شماره‌گذاری کرده و از شما می‌خواهد که هر بار دو محل را به عنوان یک زوج مرتب در نظر بگیرید و این زوج مرتب‌ها را با توجه به اینکه اختلاف مسافت کدام یک کمتر است مرتب کنید. فرض بر این است که پشنج هر روز مسیر سراسری را طی می‌کند.

ورودی

در خط اول n که تعداد محل‌هاست داده می‌شود و در خط بعد n عدد در یک خط به شما داده می‌شود که عدد i ام نشان دهنده a_i است که نمایانگر مسافت محل‌هاست. فاصله بین دو محل قدر مطلق اختلاف مسافت محل‌هاست. این نکته را در نظر داشته باشید که در صورتی که دو فاصله $|x - y|$ و $|z - t|$ با هم برابر بود، زوجی اولویت دارد که مسافت یکی از تک‌محل‌های این زوج از هر دو تک‌محل دیگر کمتر باشد. به طور مثال اگر x از z و t کوچک‌تر بود، زوج (x, y) اولویت دارد.

توجه کنید که ممکن است مسافت یک محل چندین بار در ورودی بیاید و شما نباید مقدار تکراری را در نظر بگیرید؛ بنابراین باید از هر مقدار فقط یکی را در نظر بگیرید.

$$1 \leq n \leq 90$$

$$-2\,000 \leq a_i \leq 2\,000$$

خروجی

شما باید در $\binom{m}{2}$ خط خروجی، زوج مرتب $[x, y]$ را خروجی بدهید، به طوری که $x < y$. این زوج مرتبها بر اساس قدرمطلق اختلاف x و y مرتب شده‌اند. توجه کنید که m برابر با تعداد اعداد متمایز آرایه a است. در صورتی که اختلافها برابر بود، بر اساس مقدار x مرتب کنید.

مثال

ورودی نمونه ۱

4
-1 -2 3 6

خروجی نمونه ۱

[-2, -1]
[3, 6]
[-1, 3]
[-2, 3]
[-1, 6]
[-2, 6]

به طور مثال، در اینجا، اگر اعداد را مرتب‌شده در نظر بگیریم، بازه اعداد به صورت رو به رو خواهد بود:

6, 3, -1, -2

حال این بازه‌ها که سر و ته هر یک زوج مرتبی از اعداد بالاست را در نظر بگیرید:

[6, 3], [6, -1], [3, -1], [6, -2], [3, -2], [-1, -2]

اختلاف سر و ته هر بازه به صورت رو به رو خواهد بود:

3, 7, 4, 8, 5, 1

اگر این بازه‌ها را بر اساس اختلاف سر و ته هر بازه مرتب کنیم به خروجی می‌رسیم.

ورودی نمونه ۲

7

22 43 32 6 30 41 43

خروجی نمونه ۲

[30, 32]

[41, 43]

[22, 30]

[32, 41]

[22, 32]

[30, 41]

[32, 43]

[30, 43]

[6, 22]

[22, 41]

[22, 43]

[6, 30]

[6, 32]

[6, 41]

[6, 43]

در این ورودی فقط یکی از ۴۳ ها در نظر گرفته شده است.