

دستگاه فروش گیاه

- محدودیت زمان: 2 ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

زهرة یک ایده جدید برای کسب و کارش پیدا کرده است: دستگاهی برای فروش گیاهان! او دستگاهی با n قفسه خریداری کرده و k گیاه مختلف دارد که هر گیاه i با نوع گیاه b_i و قیمت c_i مشخص شده است. او می‌تواند هر تعداد گیاه را در هر قفسه قرار دهد، اما تمام گیاه‌های یک قفسه باید از یک نوع باشند. زهرة مطمئن است که تمامی گیاه‌هایی که در قفسه‌های دستگاه قرار می‌دهد به فروش خواهند رفت. بنابراین، از شما خواسته است که حداکثر درآمدی که می‌تواند از فروش این گیاهان به دست آورد را محاسبه کنید.

ورودی

خط اول هر ورودی شامل دو عدد صحیح n و k است، که n تعداد قفسه‌ها در دستگاه و k تعداد گیاه‌های موجود برای زهرة است.

$$1 \leq n, k \leq 10^5$$

در k خط بعدی، هر خط شامل دو عدد صحیح b_i و c_i است ($1 \leq b_i \leq k, 1 \leq c_i \leq 1000$) (نوع گیاه و قیمت گیاه i).

$$1 \leq b_i \leq k$$

$$1 \leq c_i \leq 1000$$

خروجی

یک عدد صحیح چاپ کنید که حداکثر درآمدی که زهرة می‌تواند از فروش گیاهان به دست آورد را نشان می‌دهد.

مثال

ورودی نمونه ۱

3 3
2 6
2 7
1 15

خروجی نمونه ۱

28

زهرة ۳ قفسه در دستگاه فروش گیاهان دارد. او می‌تواند، به عنوان مثال، دو گیاه از نوع ۲ را روی قفسه اول و یک گیاه از نوع ۱ را روی قفسه دوم قرار دهد. در این صورت، مجموع قیمت گیاه‌ها برابر خواهد بود با $28 = 15 + 7 + 6$.

ورودی نمونه ۲

1 3
2 6
2 7
1 15

خروجی نمونه ۲

15

او تنها یک قفسه دارد. بهترین گزینه این است که یک گیاه از نوع ۱ را روی آن قرار دهد. در این صورت، مجموع قیمت برابر با ۱۵ خواهد بود.

دشت خورشید

- محدودیت زمان: 2 ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

در سرزمینی به نام دشت خورشید، n برج کوچک کنار هم در یک صف ساخته شده‌اند. هر برج در ابتدا ارتفاع (h) مشخصی دارد که با یک عدد مثبت اندازه‌گیری می‌شود. هر روز، موج عظیمی از آب از سمت چپ دشت به برج‌ها برخورد می‌کند و باعث می‌شود ارتفاع برخی از برج‌ها کاهش یابد.

به طور دقیق تر، هر روز و برای هر برج از ابتدا تا انتها، موج شرایط زیر را ایجاد می‌کند:

اگر این برج آخرین برج ردیف باشد $(i=n)$ یا ارتفاع آن از برج بعدی بیشتر باشد $(h_i > h_{i+1})$ ، ارتفاعش به مقدار $\max(0, h_i - 1)$ تغییر می‌کند.

هدف این است که بفهمیم چند روز طول می‌کشد تا ارتفاع تمام برج‌ها به صفر برسد؟

ورودی

خط اول ورودی تعداد برج‌ها، سپس خط بعدی شامل ارتفاع هر کدام از برج‌ها به ترتیب است.

$$1 \leq n \leq 10^5$$

$$1 \leq h_i \leq 10^9$$

خروجی

باید تعداد روزهایی که طول می‌کشد تا ارتفاع تمام برج‌ها به صفر برسد را محاسبه کنید و به عنوان خروجی نمایش دهید.

مثال

ورودی نمونه 1

3
1 1 2

خروجی نمونه 1

4

در اینجا ارتفاع برج ها به صورت زیر تغییر میکند:

$[1, 1, 2] \rightarrow [1, 1, 1] \rightarrow [1, 1, 0] \rightarrow [1, 0, 0] \rightarrow [0, 0, 0]$

ورودی نمونه 2

2
3 1

خروجی نمونه 2

3

در اینجا ارتفاع برج ها به صورت زیر تغییر میکنند:

$[3, 1] \rightarrow [2, 0] \rightarrow [1, 0] \rightarrow [0, 0]$

ورودی نمونه 3

5
7 4 4 3 2

خروجی نمونه 3

7

بهروز - املاکی

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

بهروز، ثروتمندترین (و وسواسی ترین) املاکی خاورمیانه تصمیم گرفته است تا عملیات خود را گسترش دهد. در راستای این هدف، بهروز به دنبال خرید جذابترین ملک در سرزمین سلطان است. در نظر بهروز ملکی جذاب است که :

۱. بیشترین مساحت را داشته باشد

۲. مربعی باشد

حال بهروز نقشه‌ای از سرزمین سلطان در اختیار دارد، اما با توجه به سالیان سال دوری از برنامه نویسی سواد نوشتن کدی که بتواند این ملک را (به صورت بهینه) پیدا کند ندارد. برنامه‌ای بنویسید که با گرفتن یک ماتریس m در n مساحت بزرگترین ملک مربعی و همینطور مختصات نقطه شروع آن را بدهد.

ورودی

در خط اول ورودی دو عدد طبیعی m و n که به ترتیب نشان دهنده تعداد ردیف ها و تعداد ستون های نقشه هستند داده می‌شود.

$$1 \leq m, n \leq 2000$$

در ادامه یک ماتریس m در n که هر عنصر آن می‌تواند 0 (وجود نداشتن ملک در عنصر) یا 1 (وجود ملک در عنصر) باشد داده می‌شود.

خروجی

در یک خط به ترتیب مساحت ملک، ردیف شروع ملک اول و ستون شروع ملک اول را چاپ کنید.

مثال

ورودی نمونه ۱

```

5 4
0 0 0 1
1 1 0 0
1 1 1 1
0 1 1 1
0 0 0 0

```

خروجی نمونه ۱

```

4 1 0

```

طول ضلع بزرگترین ملک ۲ است و مساحت آن ۴ است. همچنین ملک با مختصات شروع (۱,۰) اولین ملکی است که مساحت ۴ دارد.

ورودی نمونه ۲

```

3 6
0 0 0 1 1 1
1 1 1 0 0 0
0 0 0 1 1 1

```

خروجی نمونه ۲

```

1 0 3

```

ملک مربعی با مساحت بیشتر از ۱ وجود ندارد. در ضمن به اولویت ردیف به ستون دقت کنید (خانه ردیف ۲ و ستون ۱ اولین ملک نیست)

بهروز - دهخدا

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

پس از زیر و رو کردن بازار املاک در سرزمین سلطان، دیگر زمینی نمانده بود که بهروز بخرد، بنابراین بهروز با طی کردن ناگهانی طبقات هرم مازلو، نیازی عمیق به خودشکوفایی پیدا کرد. بهروز جدید، برخلاف بهروز قدیم دیگر دنبال پول و ثروت و قدرت نبود، بلکه به دنبال شناخت خود و دنیای اطراف خود و همینطور تبدیل شدن به بهترین نسخه خود بود. در این راستا بهروز تصمیم گرفته تا زبان مادری خود را که لغت نامه درست و حسابی ندارد، مثل دهخدا در یک لغت نامه (لغت نامه بهروز) گردآوری کند. مهم ترین مشکل بهروز در ایجاد این لغت نامه حفظ ترتیب واژگانی لغات است. به طور کلی رشته A باید قبل از رشته B بیاید اگر :

۱. طول A از B کمتر باشد و A یک پیشوند از B باشد (مثلا اگر A برابر ABC باشد، B می‌تواند ABCD و یا ABCA باشد)

۲. در اولین موقعیتی که A و B کاراکتر یکسانی ندارند، کاراکتر A کوچکتر از B باشد (مثلا اگر A برابر ABC باشد، B برابر ABE باشد)

مسئله ترتیب واژگان به قدری برای بهروز مهم است که حتی حاضر است برخی از کلمات زبان خود را کامل برعکس کند (!). البته برای انجام این عملیات بهروز نیاز به صرف مقداری پول وابسته به هر لغت دارد (بله در دنیای بهروز هر چیزی با پول خریدنی است). با وجود اینکه بهروز ادعا می‌کند این هزینه برای او مهم نیست، شما از چشم‌هایش می‌خوانید که می‌خواهد کمترین هزینه ممکن را برای این عمل صرف کند. مجموعه لغات را از بهروز بگیرید و با کمترین هزینه ممکن ترتیب واژگان را درست کنید.

ورودی

در خط اول ورودی عدد n که تعداد لغات داده شده توسط بهروز است وارد می‌شود.

$$2 \leq n \leq 10000$$

در خط بعدی n عدد وارد می‌شود که c_i نشان دهنده هزینه برعکس کردن رشته i ام است.

$$1 \leq c_i \leq 10^9$$

در n خط بعدی رشته های لغت نامه به ترتیب وارد می‌شوند.

خروجی

در یک خط از خروجی حداقل هزینه لازم برای ایجاد ترتیب واژگانی را چاپ کنید. اگر ترتیب به هیچ وجه قابل ایجاد نبود -۱ بنویسد.

ورودی نمونه ۱

```
4
0 0 8 6
bi
qp
bt
ya
```

خروجی نمونه ۱

```
8
```

در مثال بالا باید رشته سوم برعکس شود که هزینه این کار ۸ است.

ورودی نمونه ۲

```
3
10 200000 5
aaa
aa
a
```

خروجی نمونه ۲

-1

در نمونه بالا به هیچ شکل نمی‌توان ترتیب را برقرار کرد.

سیستم‌های جستجو متنی

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

یک شرکت که وظیفه آن پردازش و ساده‌سازی کلمات برای ساخت سیستم‌های هوشمند جستجوی متنی است، روزانه هزاران درخواست از سراسر دنیا دریافت می‌کند. هر درخواست شامل یک کلمه است که باید به ساده‌ترین و ابتدایی‌ترین شکل خود تبدیل شود. این فرآیند ساده‌سازی، به منابع پردازشی خاصی نیاز دارد، و تنها مقدار محدودی از این منابع برای هر کلمه موجود است.

به‌عنوان متخصص پردازش متن در این شرکت، مأموریت این است که کلمات با کمترین استفاده از منابع پردازشی به کوچک‌ترین نسخه لغوی ممکن تبدیل شوند. شما باید برای هر کلمه تصمیم بگیرید که چگونه و تا چه حد آن را تغییر دهید تا بهترین نتیجه ممکن حاصل شود.

قوانین کار:

۱. هر کلمه (s) از حروف کوچک لاتین (از a تا z) تشکیل شده است.
۲. برای ساده‌سازی هر حرف:
 - می‌توانید یک حرف را به هر حرف قبل‌تر در الفبای لاتین تبدیل کنید.
 - هر بار تغییر یک واحد از منابع پردازشی مصرف می‌کند.
۳. میزان منابع پردازشی برای هر کلمه محدود است.
۴. هدف این است که کلمه به کوچک‌ترین نسخه ممکن از نظر لغوی تبدیل شود.

ورودی

۱. تعداد درخواست‌ها (t): تعداد کلماتی که باید پردازش شوند

$$1 \leq t \leq 10^4$$

۲. برای هر درخواست:

• طول کلمه (n): تعداد حروف کلمه.

◦ میزان منابع موجود (k): حداکثر واحدهای پردازشی که می‌توانید استفاده کنید.

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

$$1 \leq n \leq 2 \times 10^5$$

خروجی

برای هر کلمه، نسخه ساده‌شده و بهینه آن را چاپ کنید.

مثال

ورودی نمونه

```
3
4 2
sawu
4 3
orwq
8 4
xmkewapr
```

خروجی نمونه

```
qawu
lrwq
tmketapr
```

Gold Miner (امتیازی)

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

در این سوال قرار است ورژن دیگری از Gold Miner را بازی کنیم. این ورژن به گونه ای طراحی شده است که محیط بازی یک مستطیل $n \times m$ است که از خانه‌های 1×1 تشکیل شده، همچنین در برخی خانه ها یک صندوقچه گنج پنهان شده. (در هر خانه حداکثر یک صندوقچه وجود دارد.) همچنین برای یافتن گنج یک بیل مکانیکی در اختیار داریم که میتواند مربعی به مساحت $r \times r$ حفر کند. اگر گوشه پایین سمت چپ مربع موردنظر در خانه (x, y) قرار بگیرد همه گنج‌های داخل مربع $(x + r - 1, y + r - 1)$...1) را به دست میاوریم. توجه داشته باشید که کل این مربع داخل محیط بازی قرار می‌گیرد.

روند بازی

k تا صندوقچه داریم که بسته به انتخاب شما می‌توانند در خانه‌های بازی جایگذاری شوند. سپس بیل مکانیکی خودش به صورت تصادفی شروع به حفر یک مربع میکند. پس باید مکان هر صندوقچه را به گونه ای انتخاب کنید تا بیشترین گنج ممکن را به دست بیاورید یعنی k صندوقچه را طوری در خانه‌های مجزا قرار دهید که وقتی بیل مکانیکی در یک موقعیت تصادفی بین $(m - r + 1)$ تا $(n - r + 1)$ موقعیت ممکن قرار میگیرد، میانگین تعداد گنج‌های یافته شده ماکسیمم باشد. برای فهم بهتر سوال به نمونه اول توجه کنید.

ورودی

شامل چهار عدد صحیح n, m, r, k است.

$$1 \leq n, m \leq 10^5$$

$$1 \leq r \leq \min(n, m)$$

$$1 \leq k \leq \min(n.m, 10^5)$$

خروجی

ماکسیمم میانگین ممکن برای تعداد گنج‌های یافته شده را تا شش رقم اعشار در خروجی چاپ کنید.

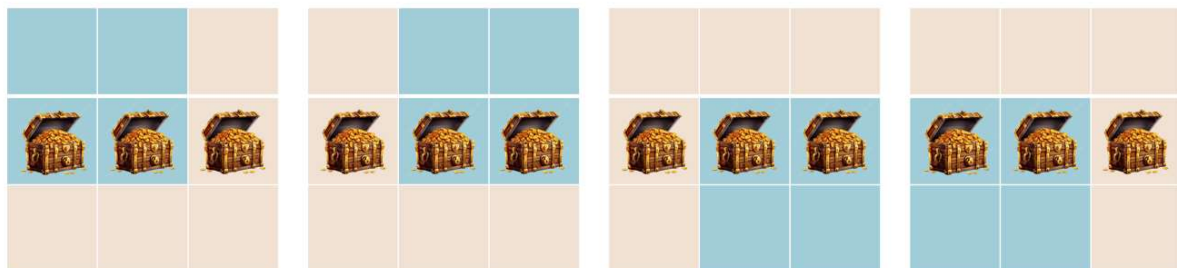
ورودی نمونه ۱

3 3 2 3

خروجی نمونه ۱

2.000000

در این مثال میتوانیم گنج‌ها را در خانه‌های $(2,1)$, $(2,2)$, $(2,3)$ قرار دهیم. در این حالت برای هر چهار موقعیت احتمالی مربع حفر شده (که با رنگ آبی مشخص شده است)، تعداد صندوقچه‌های داخل آن برابر با دو است و مقدار میانگین این حالت‌ها نیز برابر همین مقدار است.



ورودی نمونه ۲

20 59 2 88

خروجی نمونه ۲

0.319419

