

متفرق

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

علی و رضا در حال بازی تفاضل هستند، بازی به این گونه اجرا می‌شود که علی به رضا یک عدد دو رقمی می‌دهد و رضا باید رقم بزرگ‌تر این عدد را از رقم کوچک‌تر کم کند و نتیجه را بگوید. از آنجایی که رضا از دبستان کلاس‌هایش را می‌پیچاند، تفریق را به‌درستی یاد نگرفته است. اما برنامه‌نویس متبحری می‌باشد و برنامه‌ای می‌نویسد که این کار را برایش انجام دهد. حال از شما می‌خواهیم که همانند رضا این برنامه را پیاده‌سازی کنید تا شما هم به دانش "عمل تفریق" نیازی نداشته باشید.

ورودی

ورودی تنها شامل یک عدد طبیعی دو رقمی است.

خروجی

شما باید شامل یک عدد از بین ۰ تا ۹ را چاپ کنید.

مثال

ورودی نمونه ۱

خروجی نمونه ۱

6

ورودی نمونه ۲

66

خروجی نمونه ۲

0

ورودی نمونه ۳

85

خروجی نمونه ۳

3

طبقه چندم؟

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۵۱۲ مگابایت

علی و رضا که در یک ساختمان، همکار هستند؛ از طبقه‌ی همکف سوار آسانسور می‌شوند تا به طبقه‌ای که محل کارشان در آن قرار دارد بروند. ولی هنگامی که سوار آسانسور می‌شوند؛ تصمیم به انجام بازی هیجان‌انگیزی می‌گیرند!

آسانسور ساختمان آن‌ها بدین صورت است که فقط دکمه‌ی بالا و پایین و بازکردن در را دارد، دکمه‌ی بالا، آسانسور را به یک طبقه بالاتر می‌برد و دکمه‌ی پایین، آسانسور را به یک طبقه پایین‌تر.

بازی به این صورت است که این دو نفر، ۶ بار فرصت دارند تا دکمه‌ی پایین و بالای این آسانسور را فشار دهند. در نهایت، پس از اینکه ۶ بار دکمه‌های بالا و پایین آسانسور فشار داده شد؛ باید بفهمند که در کدام طبقه قرار دارند.

فرض می‌کنیم که طبقات منفی نیز داریم و آسانسور، حداکثر تا ۶ طبقه، زیر زمین نیز خواهد رفت.

حال برنامه‌ای بنویسید که با گرفتن یک رشته‌ی ۶ حرفی که شامل U به معنای رفتن به طبقه‌ی بالا و D به معنای رفتن به طبقه‌ی پایین و N به معنای بازکردن در است، شماره‌ی طبقه‌ای که در نهایت به آن می‌رویم را بازگرداند.

دقت کنید که در ابتدای حرکت، در طبقه‌ی همکف (شماره ۰) قرار داریم.

ورودی

تضمین می‌شود هیچ کاراکتر دیگری بجز U , D , N در ورودی نخواهد بود. تضمین می‌شود که هیچ فاصله ای بین کاراکتر ها نخواهد بود. تضمین می‌شود که طول ورودی ۶ خواهد بود.

خروجی

شما باید یک عدد صحیح بین -۶ تا ۶ را چاپ کنید.

مثال

در اینجا چند نمونه برای فهم بهتر صورت سوال و قالب ورودی و خروجی تست‌ها داده می‌شود.

ورودی نمونه ۱

UUUUUU

خروجی نمونه ۱

6

ورودی نمونه ۲

UDDUD

خروجی نمونه ۲

-2

ورودی نمونه ۳

UNDDUD

خروجی نمونه ۳

-1

جدا ساز

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۵۱۲ مگابایت

علی و رضا برادر هستند.

از آنجایی که علی، ازدواج کرده است؛ عاشق اعداد زوج و از آنجایی که رضا مجرد است؛ عاشق اعداد فرد است.

حال به این دو برادر، لیستی از اعداد صحیح داده می‌شود و از آنجایی که هر کدام از آن‌ها به یک نوع از اعداد زوج یا فرد علاقه‌مند است؛ تصمیم می‌گیرند که علی، اعداد زوج و رضا نیز اعداد فرد را برای خود جدا کنند.

از آنجایی که این دو برادر درگیر کارهای بنیادین دیگری هستند، از شما می‌خواهند تا این کار را برای آن‌ها انجام دهید.

برنامه خود را به نحوی بنویسید که این کار را برای این دو انجام دهد.

ورودی

ورودی شامل یک خط است که در آن چند عدد صحیح با فاصله از هم آمده اند.

- تضمین می‌شود که حداقل یک عدد در ورودی وجود دارد.

خروجی

خروجی شامل دو خط است که در خط اول اعداد علی به فرمت ورودی و در خط دوم اعداد رضا به فرمت ورودی نوشته شده اند.

- دقت کنید اگر لیستی هیچ عددی نداشت [] در آن خط نشان داده می‌شود.
- اعداد داخل لیست‌ها، باید به همان ترتیبی که در لیست اولیه قرار داشته‌اند، قرار بگیرند.

مثال

در اینجا چند نمونه برای فهم بهتر صورت سوال و قالب ورودی و خروجی تست‌ها داده می‌شود.

ورودی نمونه ۱

-3 -2 -1 0 1 2 3

خروجی نمونه ۱

-2 0 2
-3 -1 1 3

ورودی نمونه ۲

1 11 5 7 3

خروجی نمونه ۲

[]
1 11 5 7 3

در اینجا چون اولین لیست هیچ کاراکتری نداشت [] نشان داده می‌شود.

مقایسه کننده

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۵۱۲ مگابایت

علی و رضا که با یکدیگر برادر هستند، تصمیم گرفتند یک بازی هیجان‌انگیز دو نفره‌ای را انجام دهند تا شاید آدرنالین خون‌شان افزایش پیدا کند.

بازی از این قرار است که علی و رضا، هر کدام یک رشته انتخاب می‌کنند. سپس حرف اول رشته‌ای که حرف اولش از لحاظ لغت‌نامه‌ای کوچک‌تر است را حذف (اگر حرف اول دو رشته برابر بودند، حرف اول هر دو رشته را حذف می‌کنند.) و هر دو رشته را معکوس می‌کنند. این کار را تا جایی انجام می‌دهند که یکی از دو رشته یا هر دو رشته، خالی شوند.

حال شما باید برنامه خود را به گونه‌ای طراحی کنید که:

- اگر در نهایت یکی از دو رشته خالی شده بود، محتویات رشته‌ی دیگر را بازگرداند.
- اگر در نهایت هر دو رشته خالی شده بودند؛ عبارت `Both strings are empty!` را بازگرداند.

توجه کنید که اگر یکی از رشته‌ها خالی شد؛ رشته‌ی دیگر، نباید مجدداً معکوس شود.

ورودی

ورودی تنها شامل یک خط است که در آن دو کلمه با یک فاصله آمده است.

- بجای ورودی رشته خالی یک - می‌نشیند.

خروجی

خروجی برنامه‌ی شما باید شامل یک کلمه یا عبارت Both strings are empty! باشد.

مثال

ورودی نمونه ۱

ali salib

خروجی نمونه ۱

las

ابتدا حرف اول هر دو رشته بررسی می‌شود. چون a از s کوچکتر است، پس a از ali حذف شده و ali به li تبدیل می‌شود. سپس هر دو رشته معکوس (li به il و salib به bilas) می‌شوند.

حال از بین دو رشته‌ی جدید، چون b از i کوچکتر است، پس b از bilas حذف شده و bilas به ilas تبدیل می‌شود. سپس هر دو رشته معکوس (il به li و ilas به lasi) می‌شوند.

حال مجدداً از بین دو رشته‌ی جدید، چون l از s کوچکتر است، پس l از lasi حذف شده و lasi به li تبدیل می‌شود. سپس هر دو رشته معکوس (i به i و lasi به ilas) می‌شوند.

در نهایت، از بین ۲ رشته‌ی جدید، چون i و i با یکدیگر برابر هستند، پس هر دو i حذف می‌شود و i به یک رشته‌ی خالی و ilas به las تبدیل می‌شوند و چون یکی از رشته‌ها، خالی شده‌اند، محتوای رشته‌ی دیگر (که برابر las است) بازگردانده می‌شود.

ورودی نمونه ۲

amin nima

خروجی نمونه ۲

Both strings are empty!

ابتدا حرف اول هر دو رشته بررسی می‌شود. چون a از n کوچکتر است، پس a از amin حذف شده و amin به min تبدیل می‌شود. سپس هر دو رشته معکوس (min به nim و nima به amin) می‌شوند.

حال از بین دو رشته‌ی جدید، چون a از n کوچکتر است، پس a از amin حذف شده و amin به min تبدیل می‌شود. سپس هر دو رشته معکوس (nim به min و min به nim) می‌شوند.

حال مجدداً از بین دو رشته‌ی جدید، چون m از n کوچکتر است، پس m از min حذف شده و min به in تبدیل می‌شود. سپس هر دو رشته معکوس (in به ni و nim به min) می‌شوند.

دوباره از بین دو رشته‌ی جدید، چون m از n کوچکتر است، پس m از min حذف شده و min به in تبدیل می‌شود. سپس هر دو رشته معکوس (in به ni و in به ni) می‌شوند.

سپس از بین دو رشته‌ی جدید، چون i از n کوچک‌تر است، پس i از in حذف شده و in به n تبدیل می‌شود. سپس هر دو رشته معکوس (n به n و ni به in) می‌شوند.

مجدداً از بین دو رشته‌ی جدید، چون i از n کوچک‌تر است، پس i از in حذف شده و in به n تبدیل می‌شود. سپس هر دو رشته معکوس (n به n و n به n) می‌شوند.

در نهایت، از بین ۲ رشته‌ی جدید، چون n و n با یکدیگر برابر هستند، پس هر دو n حذف می‌شود و n به یک رشته‌ی خالی و n دوم نیز به یک رشته‌ی خالی تبدیل می‌شوند و چون هر دو رشته، خالی شده‌اند، عبارت `Both strings are empty!` بازگردانده می‌شود.

لغت نامه

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

علی و رضا که از بازی تفاضل خسته شده بودند، بازی جدیدی را شروع کردند.

بازی کلمه‌ها به این صورت است که علی رشته‌ای را به رضا می‌دهد و رضا باید کلمات خوب آن به همراه تعداد تکرارهایش را به فرهاد بدهد.

در واقع علی به رضا یک رشته می‌دهد که کلمات آن‌ها با کاراکترهای *Whitespace* از هم جدا شده‌اند؛ برای مثال اگر متن ورودی `Hello SAlIB!` باشد، کلمات آن `Hello` و `SAlIB!` هستند.

از نظر علی هر کاراکتری که جز حروف انگلیسی نباشد، یک حرف بد است و باید حذف شود. برای مثال کاراکترهای `A` و `b` حرف خوب و کاراکتر `!` حرف بد هستند.

همچنین به یک کلمه بد می‌گوییم اگر حداقل نیمی از کاراکترهای آن بد باشند. رضا باید به ازای هر کلمه خوب، کاراکترهای بد آن را حذف کند و سپس کاراکتر اول هر کلمه را بزرگ و دیگر کاراکترها را کوچک کند. برای مثال کلمه `A!` بد است اما `A!A` خوب است.

برای مثال اگر متن ورودی به صورت زیر باشد:

```
"hEllo My FriEnDs!!! thIS i$s A tE%ST For your #p#r#o#b#l#e#m a
```

بعد از این عملیات رشته به صورت زیر در می‌آید:

Hello My Friends This Is A Test For Your A

توجه کنید که در این نمونه کلمه `#p#r#o#b#l#e#m` بد است و حذف شده و از دیگر کلمه‌ها فقط کاراکترهای خوششان باقی مانده است.

در نهایت رضا باید در قالب یک لغتنامه (*Dictionary*) که کلیدهای آن کلمات متن و مقادیر هر کلید، تعداد تکرار آن کلمه در متن نهایی است برگرداند.

شما باید برنامه ای بنویسید که با گرفتن متن ورودی، لغتنامه خواسته شده را برگرداند.

ورودی

ورودی تنها شامل یک خط است که در کلمات خوب و بد است.

- کاراکترهای قابل قبول حروف بزرگ و کوچک انگلیسی است.

خروجی

شما باید در هر خط تک تک کلمات خوب را به همراه تعداد تکرار هایش که با فاصله جدا شده اند چاپ کنید.

- در صورتی که هیچ کلمه ای نبود - را چاپ کنید.

مثال

ورودی نمونه ۱

hEl10 My FriEnDs!!! thIS is A tEsT For your #p#r#o#b#l#e#m a

خروجی نمونه ۱

A 2
For 1
Friends 1
Hello 1
Is 1
My 1
Test 1
This 1
Your 1

بازیابی داده ها

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

علی و رضا بعد از ساخت این همه برنامه اشتباهی دستشان خورد و همه فایل های سیستم را پاک کردند! لطفا به آنها کمک کنید تا از فایل های بکاپ فایل های مهم را برگردانند.

خروجی تابع پایتونی شما باید یک لیست از فایل تایپ های پیدا شده باشد. تنها فایل های زیر برای ما اهمیت دارند. آن ها را پیدا کنید.

header1 = b'\x89PNG\r\n\x1a\n'	# PNG header
header2 = b'\xff\xd8\xff'	# JPEG header
header3 = b'\x42\x4d'	# BMP header
header4 = b'\x49\x49\x2a\x00'	# TIFF header
header5 = b'\x47\x49\x46\x38'	# GIF header
header6 = b'\x50\x4b\x03\x04'	# ZIP header
header7 = b'\x7fELF'	# ELF header
header8 = b'\x25\x50\x44\x46'	# PDF header
header9 = b'\x49\x44\x33'	# MP3 header
header10 = b'\xff\xfb'	# MPEG header
header11 = b'\x00\x00\x01\x00'	# PDDF header
header12 = b'\x00\x01\x00\x00'	# ICO header

ورودی

ورودی تنها شامل یک خط با تایپ بایت است.

خروجی

خروجی برنامه‌ی شما باید شامل یک خط باشد که به ترتیب فرمت‌هایی که درون ورودی است را با یک فاصله نمایش دهد.

مثال

ورودی نمونه ۱

```
b'\x89PNG\r\n\x1a\nBMII*\x00\xff\xd8\xff\xff\xd8\xff\x00\x00\x01\x00\
```

خروجی نمونه ۱

PNG JPEG BMP TIFF PDDF ICO