

## بررسی کارت بانکی

الگوریتم luhn، الگوریتمی است که از آن برای بررسی درست بودن یک شماره کارت ورودی به یک سیستم بانکی استفاده می‌شود تا کاربر نتواند به صورت رندوم یک شماره کارت غیر معتبر را وارد سیستم کند. این الگوریتم توسط VISA Card و Master Card و برخی دیگر از بانک‌ها استفاده می‌شود.

طوری که این الگوریتم کار میکند به این صورت است که رقم‌های فرد یک ورودی را ضرب در دو می‌کند و مجموع رقم‌های حاصل را ذخیره می‌کند. سپس ارقام زوج ورودی را به حاصل کل اضافه می‌کند. در صورتی که یکان حاصل، 0 باشد، آن شماره کارت معتبر است.

برای مثال شماره کارت 12 رقمی 328100002931 را اینگونه بررسی می‌کنیم:

۱. رقم‌های فرد (از چپ) ضرب در دو:  $2 \times 3 = 6$ ;  $4 \times 2 = 8$ ;  $0 \times 2 = 0$ ;  $0 \times 2 = 0$ ;  $0 \times 2 = 0$ ;  $16 \times 2 = 32$ ;  $6 \times 2 = 12$ ;  $3 \times 2 = 6$ .  
6.

۲. مجموع ارقام حاصل:  $23 = 6 + 4 + 0 + 0 + 0 + 16 + 6 + 12$ .

۳. مجموع ارقام زوج:  $13 = 1 + 9 + 0 + 0 + 1 + 2$ .

۴. مجموع این دو عدد به دست آمده:  $36 = 13 + 23$ .

از آنجایی که 36 یکانش 0 نیست، پس این شماره کارتی معتبر نیست. اما اگر همین محاسبات به یک عددی ختم شود که یکانش 0 باشد، آن کارت، شماره کارتی معتبر است.

البته برای بررسی معتبر بودن یک کارت صرفاً بررسی این نسبت ریاضی از طریق این الگوریتم کافی نیست و باید توسط پایگاه داده متصل به بانک نیز معتبر بودن کارت بررسی شود. این نسبت ریاضی صرفاً از بررسی کارت‌های نامعتبر از پایگاه داده جلوگیری می‌کند.

## ورودی

ورودی تنها شامل یک خط است که در آن یک شماره کارت 12 رقمی یا 16 رقمی آمده است.

## خروجی

برنامه در خروجی باید در صورت معتبر بودن شماره کارت True را چاپ بگیرد و در غیر این صورت False.

## مثال

### ورودی نمونه ۱

582710372834

### خروجی نمونه ۱

False

### ورودی نمونه ۲

4503600000024514

### خروجی نمونه ۲

True

## خوانایی متن

- محدودیت زمان: 3 ثانیه
- محدودیت حافظه: 512 مگابایت

فرمول کلمن-لیا (Coleman-Liau) فرمولی است که میزان خوانایی یک متن ورودی را بررسی می‌کند. این فرمول به این شکل است:

$$0.0588 * L - 0.296 * S - 15.8$$

در این فرمول  $L$  برابر با میانگین تعداد حروف در 100 کلمه از متن است و  $S$  برابر با میانگین تعداد جملات در 100 کلمه از متن است. که اگر آنها را جایگذاری کنیم و محاسبه کنیم، جزء صحیح عدد حاصل به ما می‌گوید متن مناسب یک دانش آموز کلاس  $n$  ام است. مثلا اگر جزء صحیح حاصل 5 بود، یک دانش آموز کلاس پنجم می‌تواند آن متن را مطالعه کند.

شما باید برنامه ای بنویسید که فایل "input.txt" را بخواند و متن درون آن را با فرمول کلمن-لیا بسنجد و در خروجی جزء صحیح حاصل را چاپ بگیرد.

## تذکر

برای سادگی کار علائم نگارشی را جزو حروف کلمات در نظر بگیرید. برای مثال "bye." یک کلمه 4 حرفی محسوب می‌شود.

## ورودی

برنامه از کاربر ورودی نمی‌گیرد و فقط فایل input.txt را میخواند.

## خروجی

جزء صحیح حاصل فرمول کلمن-لیا که بیانگر آن است که متن مناسب چه مقطع تحصیلی ای است.

## مثال

### متن نمونه ۱

They followed Professor McGonagall across the flagged stone floor. Harry could

### خروجی نمونه ۱

12

### متن نمونه ۲

Harry Potter was a highly unusual boy in many ways. For one thing, he hated th

5

## یافتن پالیندروم ها

- محدودیت زمان: 3 ثانیه
- محدودیت حافظه: 512 مگابایت

با Palindrome ها آشنا هستید. به کلماتی که برعکسشان با خودشان برابر است Palindrome میگوئیم. مثل mom، dad و ...

برنامه ای بنویسید که فایل "input.txt" را بخواند و تمام کلمات Palindrome آن را بیابد و در فایل "output.txt" این کلمات را بنویسد.

## نکات

۱. در هر خط تنها یک کلمه باید قرار گیرد.
۲. کلمات تکراری نباید وجود داشته باشند.
۳. برای یکسان بودن خروجی شما با تست کیس ها، می‌توانید با تابع sorted() نتایج را مرتب کنید و سپس در فایل output.txt چاپ بگیرید.
۴. برای سادگی کار فرض می‌کنیم علائم نگارشی هم جزوی از کلمه هستند و "mom." دیگر پالیندروم نیست.

## ورودی

برنامه از کاربر ورودی نمی‌گیرد، باید فایل input.txt را بخواند.

## خروجی

فایل output.txt را به عنوان خروجی می‌دهد که باید در هر خط یک کلمه Palindrome غیر تکراری موجود در input.txt را قرار دهد.

## مثال

### متن نمونه ۱

They followed Professor McGonagall across the flagged stone floor. Harry could

### خروجی نمونه ۱

a

### متن نمونه ۲

His mom and dad should definitely get him into a high quality university, but

### خروجی نمونه ۲

a  
dad  
mom  
wow

## تبدیل هگزادسیمال

- محدودیت زمان: 3 ثانیه
- محدودیت حافظه: 512 مگابایت

اعداد هگزادسیمال، به اعداد بر مبنای 16 گفته می‌شود. در این مبنای اعداد، ارزش 10 را با A، ارزش 11 را با B و ... و ارزش 15 را با F نمایش می‌دهد. یعنی

۱.  $A = 10$

۲.  $B = 11$

۳.  $C = 12$

۴.  $A2 = (10 * 16) + (2 * 1) = 162$

۵.  $B2 = (11 * 16) + (2 * 1) = 178$

۶.  $CC = (12 * 16) + (12 * 1) = 204$

۷.  $6CA = (6 * 256) + (12 * 16) + (10 * 1) = 1738$

تابعی بنویسید که به عنوان ورودی یک عدد هگزادسیمال دریافت کند و به عدد دسیمال تبدیل کند. واضح است که حق استفاده از توابع از پیش تعیین شده پایتون برای تبدیل اعداد به مبناهای مختلف را ندارید.

## ورودی

ورودی تنها شامل یک خط است که در آن یک عدد هگزادسیمال آمده است.

## خروجی

عدد دسیمال متناظر با عدد هگزادسیمال ورودی باید چاپ شود.

## مثال

ورودی نمونه ۱

ffa70a

خروجی نمونه ۱

16754442

ورودی نمونه ۲

8FA

خروجی نمونه ۲

2298



## مرتب سازی حبابی

- محدودیت زمان: 3 ثانیه
- محدودیت حافظه: 512 مگابایت

مرتب سازی حبابی یک الگوریتم مرتب سازی است که بر روی لیست نامرتب چندین بار گذر می کند و در هر گذر بزرگترین جمله را به آخر لیست هدایت می کند. همانطور که حباب ها در آب هرچه بالاتر می آیند بزرگتر می شوند.

در این الگوریتم از حلقه های تو در تو استفاده می شود، به این صورت که باید  $n - 1$  بار لیست را پیمایش کنیم و در هر پیمایش عدد های متوالی درون لیست را با یکدیگر مقایسه می کنیم و اگر عددی از عدد جلویی اش بزرگتر بود، مکان این دو عدد را با یکدیگر جابجا می کنیم و در انتهای هر پیمایش، بزرگترین عدد موجود به انتهای لیست می رسد تا زمانی که تمام اعداد در جای درستی قرار گیرند.

برای درک بیشتر این الگوریتم می توانید راجب آن بیشتر در اینترنت مطالعه کنید.

## تذکر

در صورتی که مرتب سازی شما از min استفاده کند و یا از الگوریتم دیگری استفاده کنید، نمره این سوال برای شما 0 رد می شود.

همچنین در لیست خروجی اعداد اعشاری هم می توانند قرار گیرند اما اعداد صحیح نباید با 0 نمایش داده شوند و باید جنس آنها به صورت int حفظ شود.

## ورودی

لیستی از اعداد که با فاصله از هم جدا شده اند.

## خروجی

لیست مرتب شده.

مثال

ورودی نمونه ۱

1.5 -2 7.2 4.0 3

خروجی نمونه ۱

[-2, 1.5, 3, 4, 7.2]

## سری های مرتب

- محدودیت زمان: 3 ثانیه
- محدودیت حافظه: 512 مگابایت

تابعی بازگشتی بنویسید که تمام سری های مرتب  $k$  عدد از اعداد صحیح 1 تا  $n$  را تولید کند.

برای مثال فرض کنید میخواهیم سری های مرتب 3 عدد از اعداد صحیح 1 تا 5 را تولید کنیم. خروجی به این شکل خواهد بود:

```
1 2 3
1 2 4
1 2 5
1 3 4
1 3 5
1 4 5
2 3 4
2 3 5
2 4 5
3 4 5
```

## ورودی

ورودی تنها شامل یک خط است که در آن دو عدد طبیعی  $n$  و  $k$  با فاصله از هم آمده است.

$$1 \leq k \leq n \leq 100$$

## خروجی

خروجی شما باید در هر خط یکی ازین سری ها را چاپ بگیرد و سری ها باید به ترتیب کوچک به بزرگ باشند.

## مثال

### ورودی نمونه ۱

5 3

### خروجی نمونه ۱

1 2 3

1 2 4

1 2 5

1 3 4

1 3 5

1 4 5

2 3 4

2 3 5

2 4 5

3 4 5

### ورودی نمونه ۲

6 2

### خروجی نمونه ۲

1 2

1 3

1 4

1 5

1 6

2 3

2 4

2 5

2 6

3 4

3 5

3 6

4 5

4 6

5 6