

بازی کلمه‌ها

فرهاد و سلیب که از بازی تفاضل خسته شده بودند، بازی جدیدی را شروع کردند.

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

در واقع فرهاد به سلیب 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 فرهاد 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. سلیب 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 سلیب 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 تابعی به نام `words_check` بنویسید که با گرفتن متن ورودی، لغتنامه خواسته شده را برگرداند.

▼ نمونه

```
1 >> words_check("""hEllo My FriEnDs!!!
2 thIS is A tEsT For your #p#r#o#b#l#e#m a""")
3 {'A': 2, 'For': 1, 'Friends': 1, 'Hello': 1, 'Is': 1, 'My': 1, 'Test':
```

▼ نکات

- برای این که کلمات رشته `s` را بر حسب کاراکترهای *Whitespace* جدا کنید، می‌توانید از تابع `s.split()` استفاده کنید.
- برای ارسال باید یک فایل پایتون ارسال کنید که در آن تابع `words_check` به شکل گفته شده وجود داشته باشد.
- می‌توانید فایل اولیه‌ی خام را از [این لینک](#) دانلود کنید.