

## کار با لیست

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

برنامه‌ای بنویسید که لیستی از اعداد را دریافت کرده و تمامی صفرهای موجود در لیست را به آخر لیست منتقل کند.

### ورودی

ورودی شامل دنباله از اعداد می‌باشد که توسط کاراکتر , جدا شده‌اند، به مثال زیر توجه کنید.

6,9,4,9,3,8,9,73,7

Copy Plain text

### خروجی

خروجی لیستی از اعداد است که صفرهای موجود در آن در آخر آن لیست قرار دارند.

### مثال

به مثال‌های زیر توجه کنید.

#### ورودی نمونه ۱:

2,0,2,1,5,3,5,1,7,9

#### خروجی نمونه ۱:

[2, 2, 1, 5, 3, 5, 1, 7, 9, 0]

#### ورودی نمونه ۲:

5,7,0,5,9,1,4,3,6,2

#### خروجی نمونه ۲:

[5, 7, 5, 9, 1, 4, 3, 6, 2, 0]

## لیست تو در تو

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

برنامه‌ای بنویسید که لیستی از لیست‌ها را دریافت کرده، و طول بزرگ‌ترین و کوچک‌ترین لیست را چاپ کند.

### ورودی

ورودی به صورت لیستی از لیست‌هاست، به مثال زیر توجه کنید:

```
[[1,2],['a','b'],[1,2,3,4,5,9],[]]
```

### توجه

به کمک تابع `eval()` می‌توانید ورودی را که به صورت رشته به شما داده می‌شود را به لیست تبدیل کنید. به مثال زیر توجه کنید.

```
1 >>> lst = eval("[[1,2],['a','b'],[1,2,3,4,5,9],[]]")
2 >>> lst
3 >>> [[1,2],['a','b'],[1,2,3,4,5,9],[]]
```

### خروجی

خروجی به صورت یک tuple است، که عنصر اول آن طول کوچک‌ترین لیست و عنصر دوم آن طول بزرگ‌ترین لیست می‌باشد.

### مثال

به مثال‌های زیر توجه کنید:

ورودی نمونه ۱:

```
[[1,2],['a','b'],[1,2,3,4,5,9],[]]
```

خروجی نمونه ۱:

```
(0,6)
```

## ساخت دیکشنری

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

برنامه‌ای بنویسید که جفت‌هایی به صورت {key:value} را از ورودی دریافت کرده و برای هر کلید لیستی بسازد که شامل تمام مقادیر آن کلید است و در نهایت خروجی را به صورت دیکشنری چاپ کند.

### ورودی

ورودی شامل جفت‌هایی به صورت {key:value} در هر خط و فضای خالی (space) در آخرین خط می‌باشد. توجه کنید که ممکن است کلیدها تکرار شوند. تعداد خط‌های ورودی نامشخص و کمتر از ۱۰۰۰۰ خط می‌باشد. به مثال زیر توجه کنید:

```
a:2
c:3
c:8
c:10
b:0
k:10
L:9
```

### خروجی

خروجی به صورت دیکشنری است که برای هر کلید، لیستی از مقادیر وجود دارد.

#### توجه:

در آخر با استفاده از دستور زیر، دیکشنری خروجی را با استفاده از کلیدها مرتب کنید.

```
1 | def sort_by_keys(dct: dict) -> dict:
2 |     return {k: dct[k] for k in sorted(dct.keys())}
```

#### توجه:

همه‌ی مقادیر value عدد صحیح می‌باشند و در دیکشنری خروجی هم باید به int تبدیل شوند.

### مثال

به مثال‌های زیر توجه کنید:

#### ورودی نمونه ۱:

```
J:5
N:4
h:4
D:1
L:5
```

k:4  
G:5  
A:2  
c:5  
Y:0  
p:5  
s:0  
Q:4  
V:2  
d:4  
f:4  
O:2  
M:1  
h:3  
E:0

خروجی نمونه ۱:

{'A': [2], 'D': [1], 'E': [0], 'G': [5], 'J': [5], 'L': [5], 'M': [1], 'N': [4], 'O': [2], 'Q': [4], 'V':

ورودی نمونه ۲:

J:5  
m:4  
h:4

خروجی نمونه ۲:

{'J': [5], 'h': [4], 'm': [4]}

## تبدیل تاپل به لیست

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

برنامه‌ای بنویسید که لیستی از **tuple**ها را از ورودی دریافت کرده و آن را به لیستی از لیست‌ها تبدیل کند.

### ورودی

ورودی به صورت لیستی از **tuple**هاست، به مثال زیر توجه کنید:

```
[('Q', 3), ('J', 8), ('D', 1), ('Q', 6), ('I', 4), ('g', 3), ('c', 5), ('d', 2), ('O', 6), ('T', 0)]
```

### توجه:

برای تبدیل رشته‌ی ورودی به لیست می‌توانید همانند تمرین ۱ از تابع `eval()` استفاده کنید.

### خروجی

خروجی به صورت لیستی از لیست‌هاست.

### مثال:

#### ورودی نمونه ۱:

```
[('v', 8), ('l', 3), ('l', 4), ('W', 4), ('A', 9), ('w', 4), ('h', 5), ('c', 6), ('W', 7), ('e', 1)]
```

#### خروجی نمونه ۱:

```
[['v', 8], ['l', 3], ['l', 4], ['W', 4], ['A', 9], ['w', 4], ['h', 5], ['c', 6], ['W', 7], ['e', 1]]
```

#### ورودی نمونه ۲:

```
[('v', 8), ('l', 3), ('l', 4, 8, 0)]
```

#### خروجی نمونه ۲:

```
[['v', 8], ['l', 3], ['l', 4, 8, 0]]
```