

نمونه سوال پایانترم

مبانی کامپیوتر و برنامه سازی - زمستان ۱۴۰۳

اگر با خطا مواجه می‌شود، دلیل خطا چیست؟

```
def x(y):  
    x = 10  
    return x + y  
print(x(1))
```

اگر با خطا مواجه می‌شود، دلیل خطا چیست؟

```
def x(y):  
    x = 10  
    return x + y  
print(x(1))
```

No error. (But you shouldn't do it!)

```
A = list()
for i in range(10):
    A.append(i)
    A.pop(i)
```

```
A = list()
for i in range(10):
    A.append(i)
    A.pop(i)
```

IndexError: pop index out of range

$d = \{\}$

$d[(0, 1, 2)] = \{0, 1, 2\}$

```
d = {}
```

```
d[(0, 1, 2)] = {0, 1, 2}
```

No error.

```
d = {}  
d["0, 1, 2"] = {0, 1, 2}  
print d.keys()["0, 1, 2"][0]
```



```
d = {}  
d["0, 1, 2"] = {0, 1, 2}  
print d.keys()["0, 1, 2"][0]
```

TypeError: list indices must be integers, not str.

خروجی قطعه کد زیر چیست؟

```
my_list = [1, 2, 3, 4, 6]
for item in my_list:
    if item % 2 == 0:
        my_list.remove(item)
print(my_list)
```

خروجی قطعه کد زیر چیست؟

```
my_list = [1, 2, 3, 4, 6]
for item in my_list:
    if item % 2 == 0:
        my_list.remove(item)
print(my_list)
```

[1, 3, 6]

خروجی قطعه کد زیر چیست؟

```
my_list = [1, 2, 3, 4, 6]
for item in my_list[:]:
    if item % 2 == 0:
        my_list.remove(item)
print(my_list)
```

خروجی قطعه کد زیر چیست؟

```
my_list = [1, 2, 3, 4, 6]
for item in my_list[:]:
    if item % 2 == 0:
        my_list.remove(item)
print(my_list)
```

[1, 3]

خروجی قطعه کد زیر چیست؟

```
A = [0,0,0,1,2,3,3,0,0,1]
for x in A:
    i = A.index(x)
    if x == 0:
        A.pop(i)
print(A)
```

خروجی قطعه کد زیر چیست؟

```
A = [0,0,0,1,2,3,3,0,0,1]
for x in A:
    i = A.index(x)
    if x == 0:
        A.pop(i)
print(A)
```

[1, 2, 3, 3, 0, 0, 1]

خروجی قطعه کد زیر چیست؟

```
my_list = [1, 2, 3]
for item in my_list:
    if item == 2:
        my_list.append(2)
print(my_list)
```


خروجی قطعه کد زیر چیست؟

```
my_list = [1, 2, 3]
for item in my_list:
    if item == 2:
        my_list.append(2)
print(my_list)
```

The loop runs forever.

تابع `first_unique_char` رشته‌ای را دریافت می‌کند و اندیس اولین کاراکتر منحصر بفرد در رشته را برمی‌گرداند. اگر چنین کاراکتری وجود نداشته باشد، تابع مقدار `-1` برمی‌گرداند. متاسفانه موقع کپی پست کردن تابع، فاصله از سر خط کپی نشده و کد بصورت زیر درآمده. برای هر خط مشخص کنید چند کاراکتر tab باید در ابتدای آن درج شود.

```
def first_unique_char(s):  
    char_count = {}  
    for char in s:  
        if char in char_count:  
            char_count[char] += 1  
        else:  
            char_count[char] = 1  
    for i in range(len(s)):  
        if char_count[s[i]] == 1:  
            return i  
    return -1
```

لیست G شامل نمرات دانشجویان است. هر عنصر لیست یک تاپل با سه مولفه است. مولفه اول شماره دانشجویی، مولفه دوم اسم درس و مولفه سوم نمره درس (از نوع int) است. با فرض اینکه دانشجو درس را فقط یک بار انتخاب کرده است، درسی را پیدا کنید که بیشترین طرفدار را داشته است.

```
participation = dict()
for x in G:
    if x[1] is in participation:
        ..... += 1
    else:
        participation[x[1]] = .....

def fun(z):
    return .....

ranking = sorted(participation.items(), key=fun, reverse=True)
print(.....)
```

لیست G شامل نمرات دانشجویان است. هر عنصر لیست یک تاپل با سه مولفه است. مولفه اول شماره دانشجویی، مولفه دوم اسم درس و مولفه سوم نمره درس (از نوع int) است. با فرض اینکه دانشجو درس را فقط یک بار انتخاب کرده است، درسی را پیدا کنید که بیشترین طرفدار را داشته است.

```
participation = dict()
for x in G:
    if x[1] is in participation:
        participation[x[1]] +=1
    else:
        participation[x[1]] = 1

def fun(z):
    return z[1]

ranking = sorted(participation.items(), key=fun, reverse=True)
print(ranking[0][0])
```

دو دانشجو را پیدا کنید که بیشترین درس را با هم داشته‌اند.

```
taken = dict()
for x in G:
    if x[0] is in taken:
        taken[.....].add(x[1])
    else:
        taken[x[0]] = .....

T = [ ..... for s1 in taken for s2 in taken if s1 != s2]

def fun(x):
    return .....

ranking = sorted(T, key=fun, reverse=True)
print(ranking[0])
```

دو دانشجو را پیدا کنید که بیشترین درس را با هم داشته‌اند.

```
taken = dict()
for x in G:
    if x[0] is in taken:
        taken[x[0]].add(x[1])
    else:
        taken[x[0]] = {x[1]}

T = [(s1,s2,len(taken[s1] & taken[s2])) \
for s1 in taken for s2 in taken if s1 != s2]

def fun(x):
    return x[2]

ranking = sorted(T, key=fun, reverse=True)
print(ranking[0])
```

قطعه کد زیر چه کاری انجام می‌دهد؟ برای $a = [3, 4, 1, 5, 2]$ چند بار متغیر temp مقداردهی می‌شود؟

```
for i in range(len(a)):
    for j in range(i+1, len(a)):
        if(a[i] < a[j]):
            temp = a[i]
            a[i] = a[j]
            a[j] = temp
```

قطعه کد زیر چه کاری انجام می‌دهد؟ برای $a = [3, 4, 1, 5, 2]$ چند بار متغیر temp مقداردهی می‌شود؟

```
for i in range(len(a)):
    for j in range(i+1, len(a)):
        if(a[i] < a[j]):
            temp = a[i]
            a[i] = a[j]
            a[j] = temp
```

sorts the list a in reverse order.

5 times.

$[5, 3, 1, 4, 2]$ — 2

$[5, 4, 1, 3, 2]$ — 1

$[5, 4, 3, 1, 2]$ — 1

$[5, 4, 3, 2, 1]$ — 1

فرض کنید $N_{p,q}$ مجموعه اعداد طبیعی بزرگتر از p و کمتر q باشد. تابع $\text{find_missing_number}$ لیست L را دریافت می کند که دقیقا همان اعداد داخل $N_{p,q}$ است با این تفاوت که یکی از اعضای $N_{p,q}$ در آن حضور ندارد. تابع زیر عدد غایب را پیدا می کند و برمی گرداند. با توجه به این توصیف، پیاده سازی تابع $\text{find_missing_number}$ را کامل کنید.

```
def find_missing_number(p,q, L):  
    s1 = -----  
    s2 = -----  
    s3 = s2 - s1  
    a = sum(L)  
    return a - s3
```