

# 总结



代码编写好后必须要经过反复测试，不能满足于几次测试结果正确。

```
In [8]: x = [1,2,1,2,1,2,1,2,1]

for i in x:
    if i == 1:
        x.remove(i)
print(x)

[2, 2, 2, 2]
```

```
In [5]: xx = [1,2,1,2,1,1,1]

for i in xx:
    if i == 1:
        xx.remove(i)
print(xx)

[2, 2, 1]
```

出现这个问题的原因是列表的自动内存管理功能。

在删除列表元素时，Python会自动对列表内存进行收缩并移动列表元素以保证所有元素之间没有空隙，增加列表元素时也会自动扩展内存并对元素进行移动以保证元素之间没有空隙。每当插入或删除一个元素之后，该元素位置后面所有元素的索引就都改变了。

```
In [7]: x = [1,2,3,4,5,6,7]

for i in x:
    print('i:', i)
    print(x.index(i))
    x.remove(i)

print(x)

i: 1
0
i: 3
1
i: 5
2
i: 7
3
[2, 4, 6]
```

```
In [ ]: #从后往前删
x = [1,2,1,2,1,1,1,2,2]
for i in range(len(x)-1,-1,-1):
    if x[i]==1:
        del x[i]
print(x)
```

```
In [ ]:
```