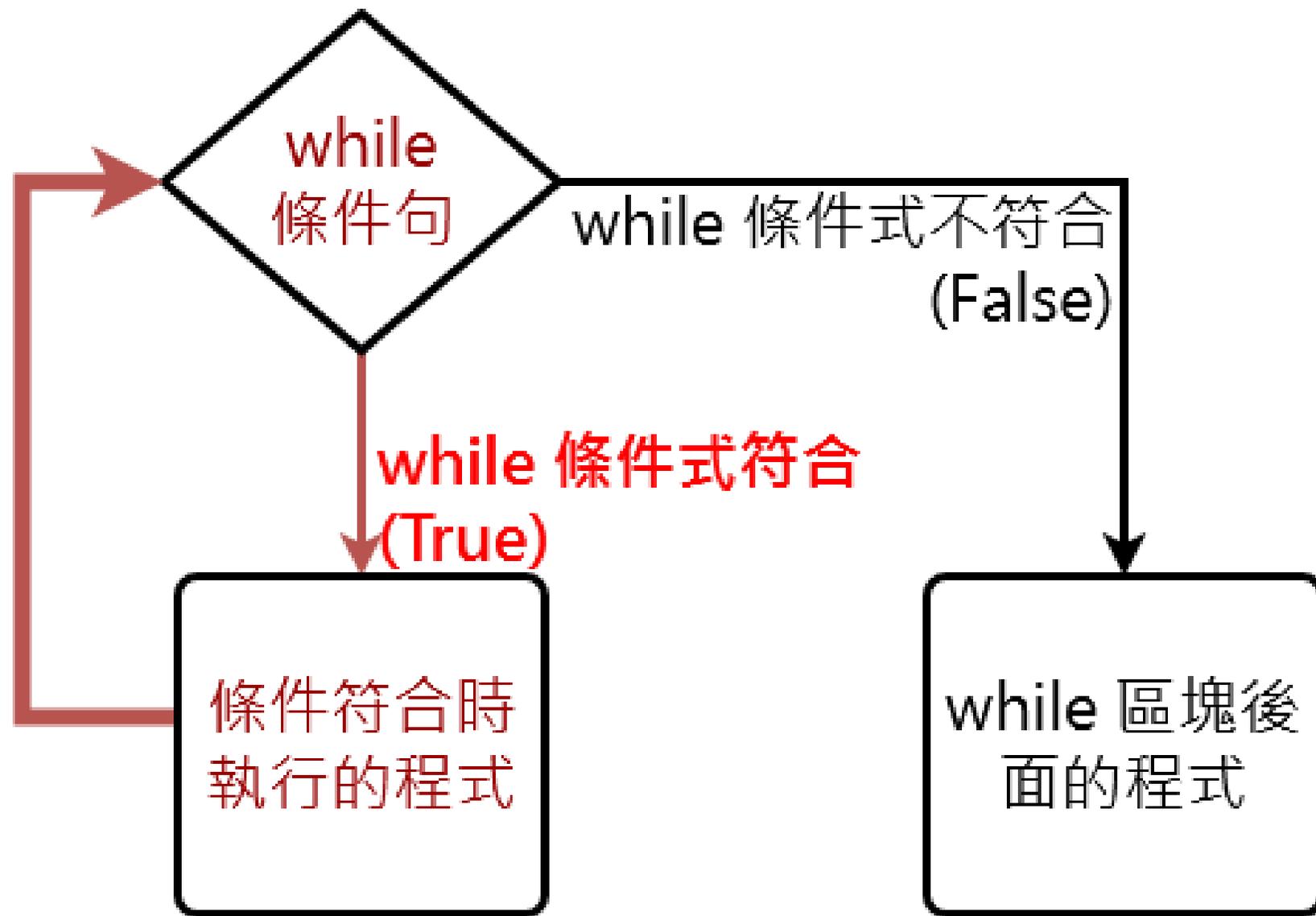


Python

if與迴圈的應用

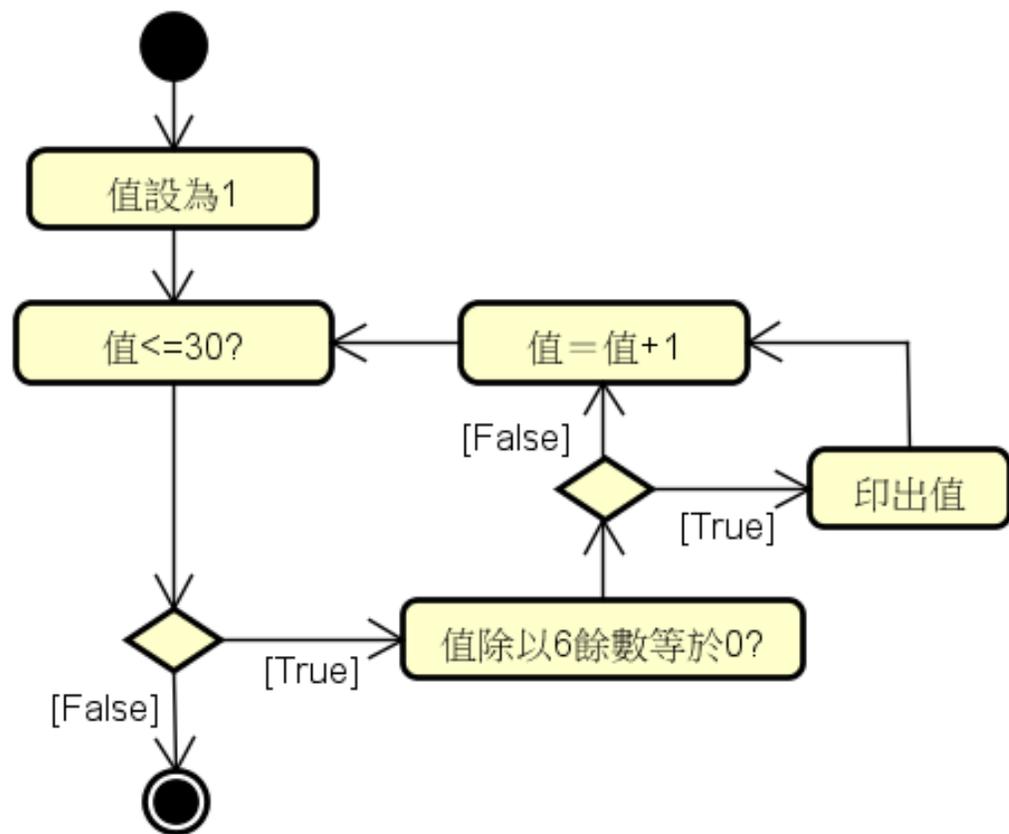
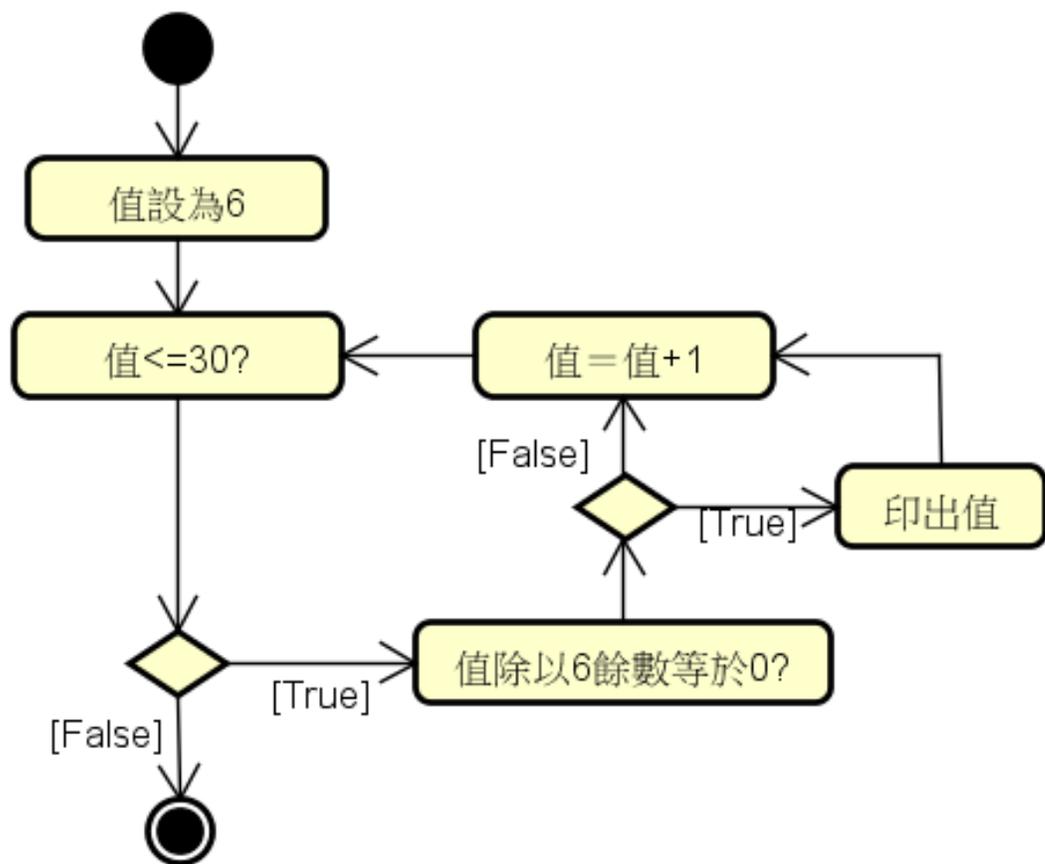


請撰寫一程式，
印出1到30所有6的倍數。

問題



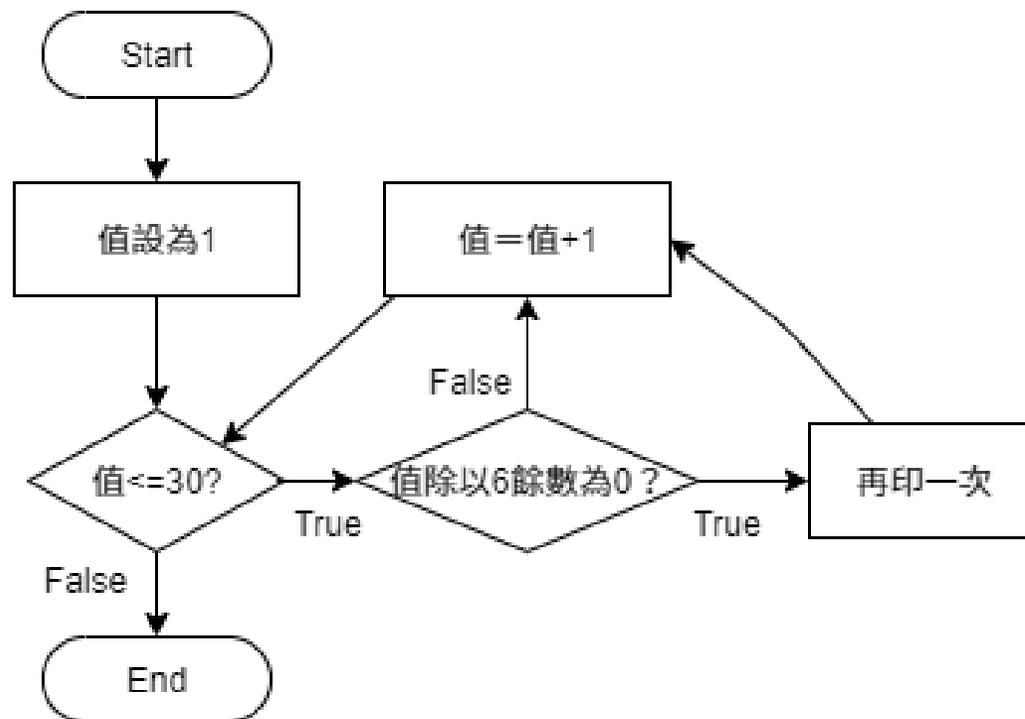
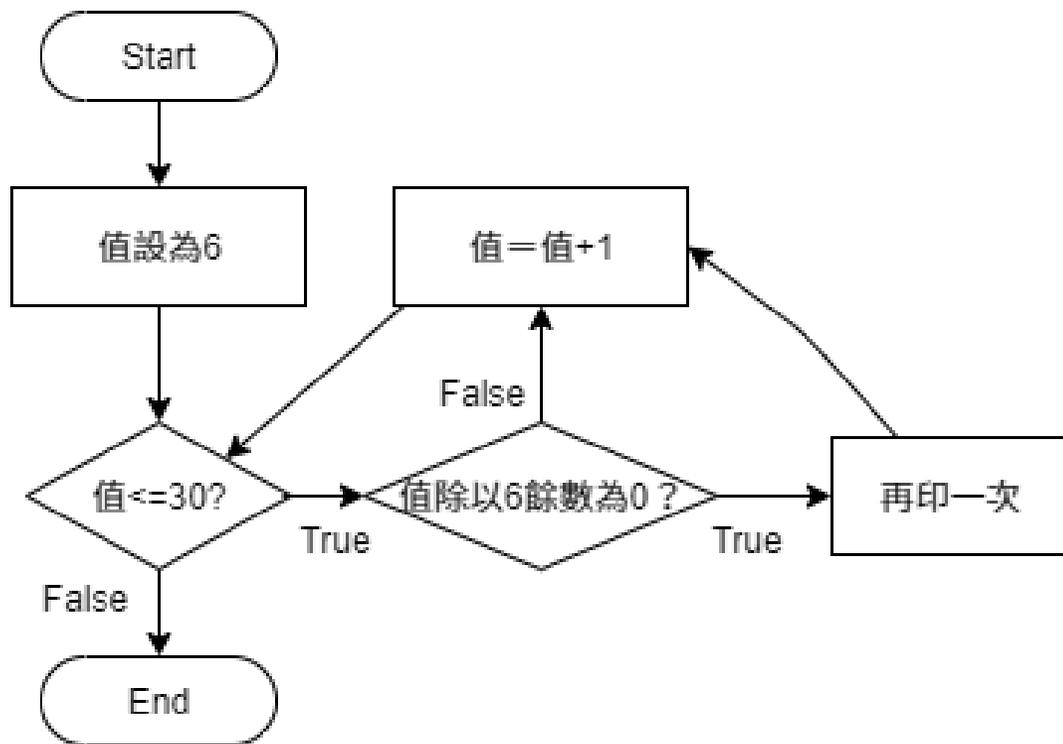
流程



UML流程圖

另一種想法

流程



另一種想法

傳統流程圖

只挑我要的!

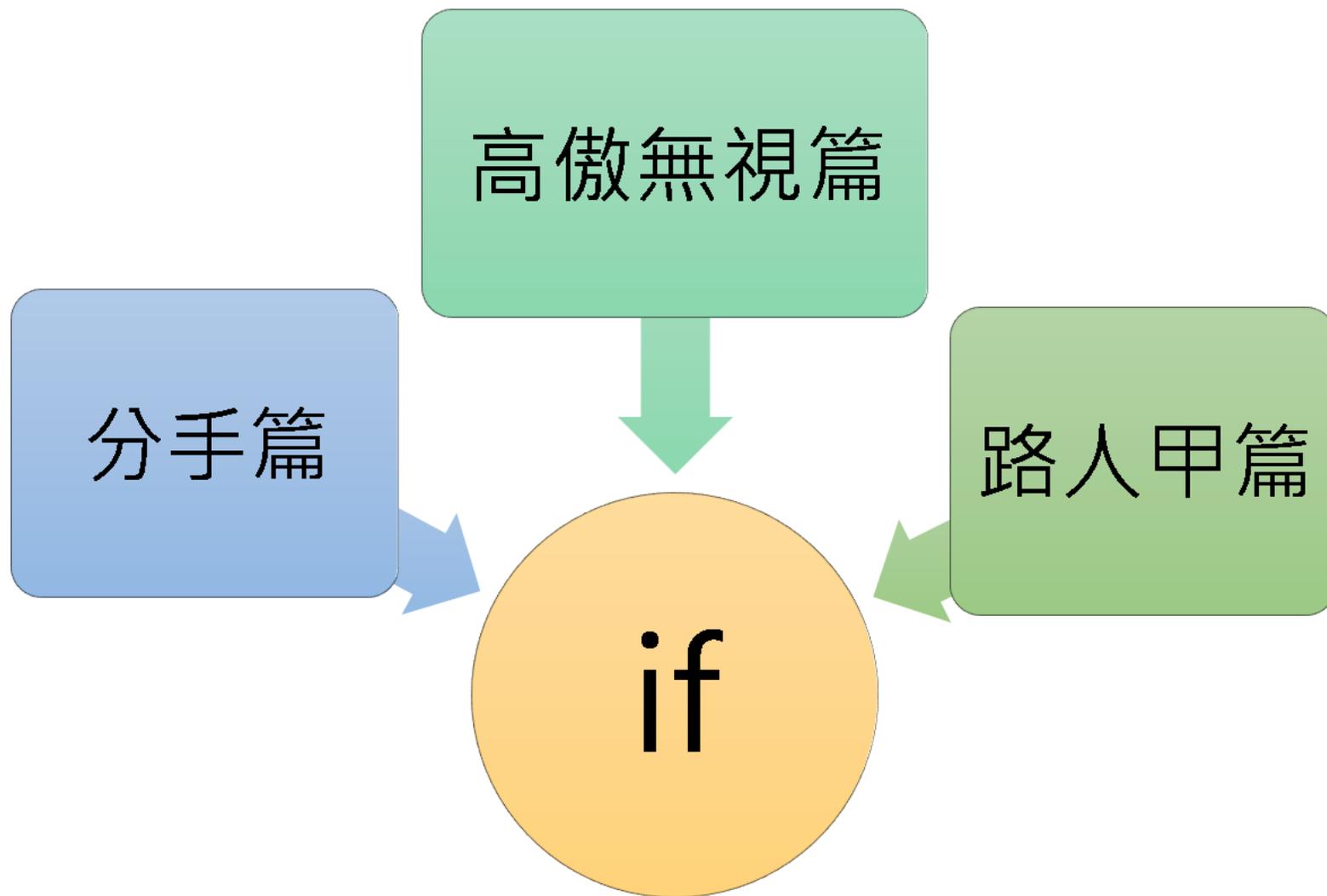
```
i = 6  
while i <= 30:  
    if i % 6 == 0  
        print(i)  
    i = i + 1
```



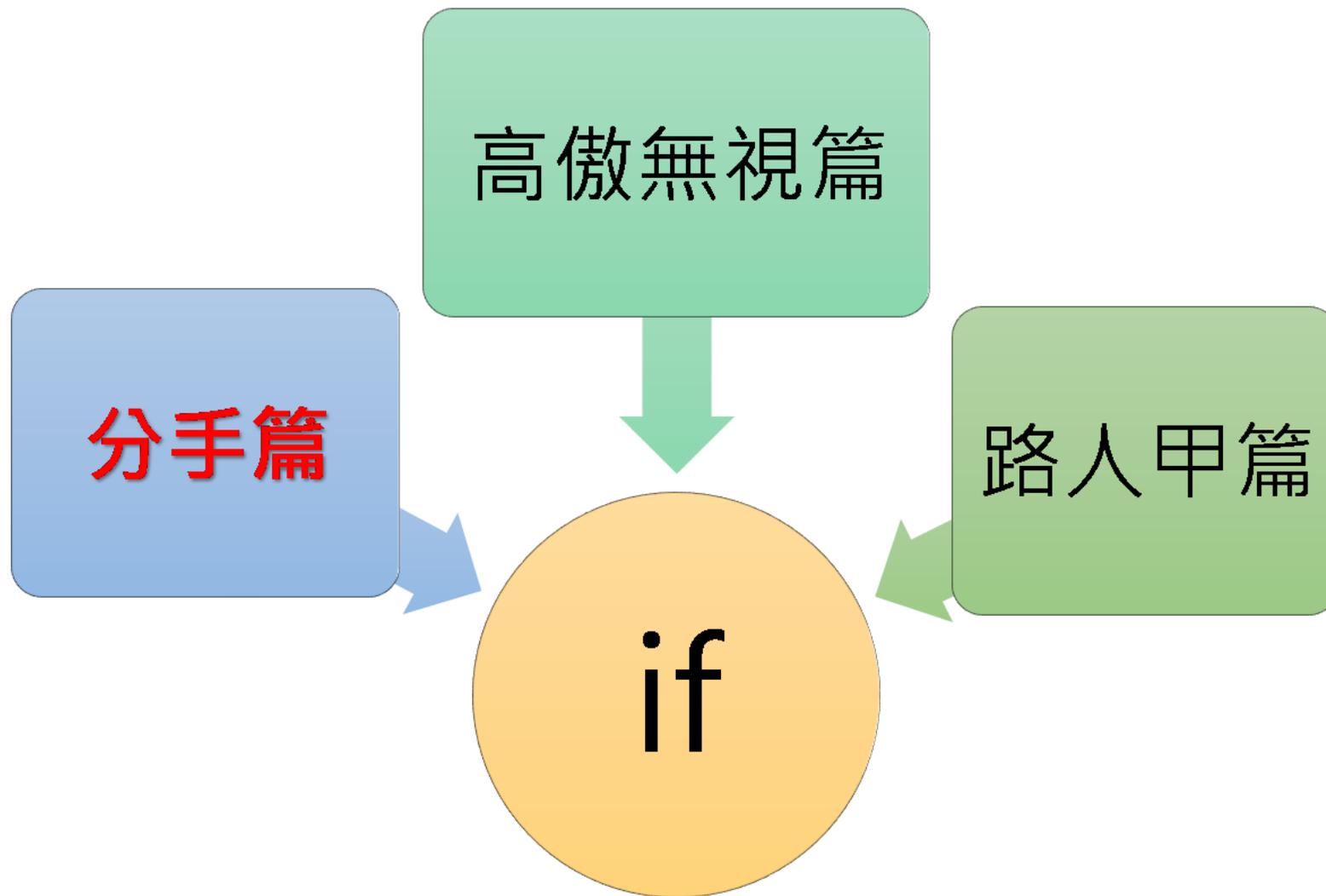
執行結果

```
6  
12  
18  
24  
30
```

if 小劇場很搶戲



if 小劇場很搶戲 – 分手篇



if 小劇場很搶戲 – 分手篇 break

```
i = 100
while i <= 130:
    if i % 6 == 0:
        print(i)
    if i % 23 == 0:
        break
    i = i + 1
```

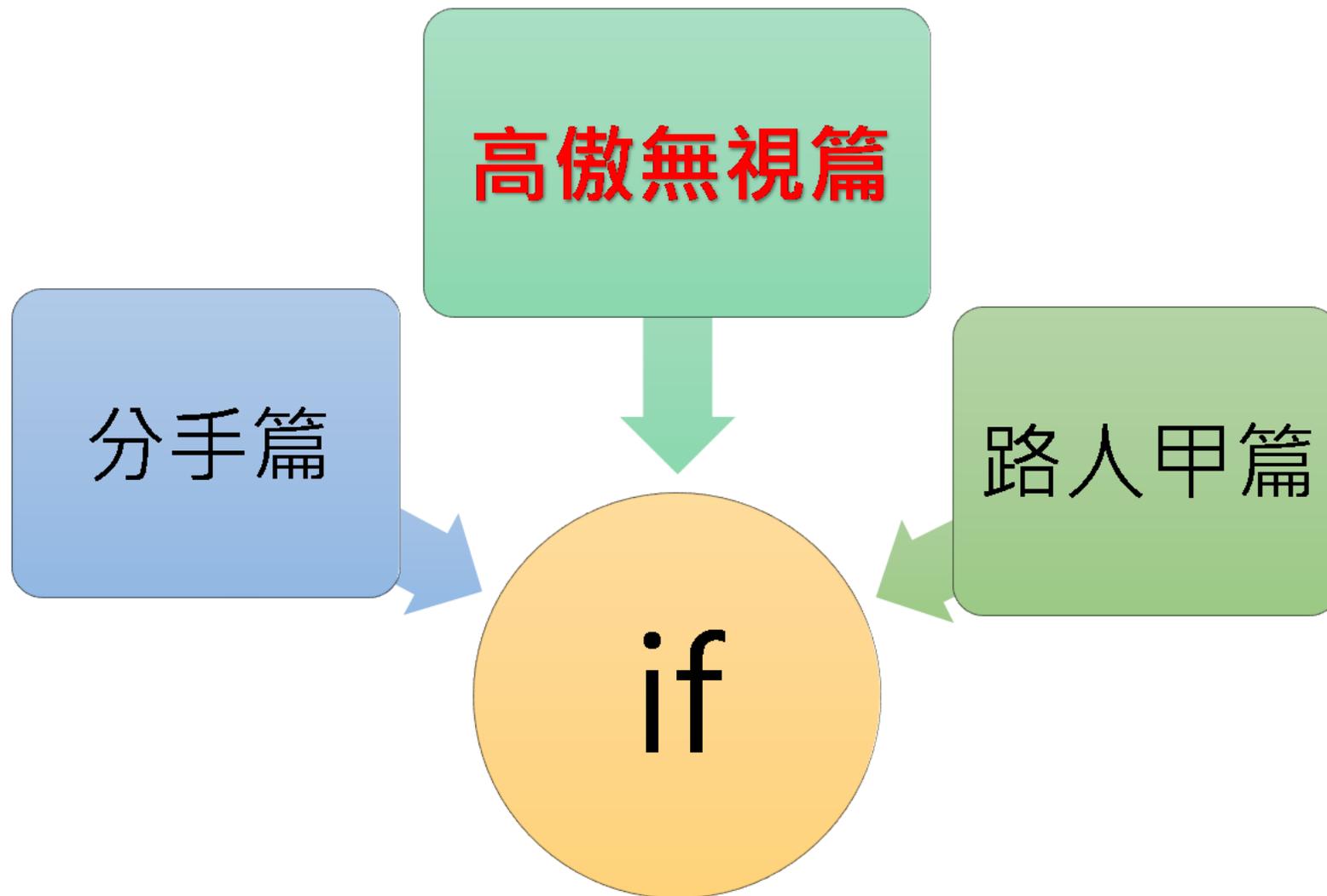
一發現不喜歡的，
就拍拍屁股走人囉！

執行 break 後，就會立刻跳出 while 迴圈，後續的 while 迴圈就不會執行

執行結果

```
102
108
114
```

if 小劇場很搶戲 – 高傲無視篇



if 小劇場很搶戲 – 高傲無視篇 continue

```
i = 1
while i <= 30:
    i = i + 1
    if i % 8 == 0:
        continue
    if i % 6 == 0:
        print(i)
```

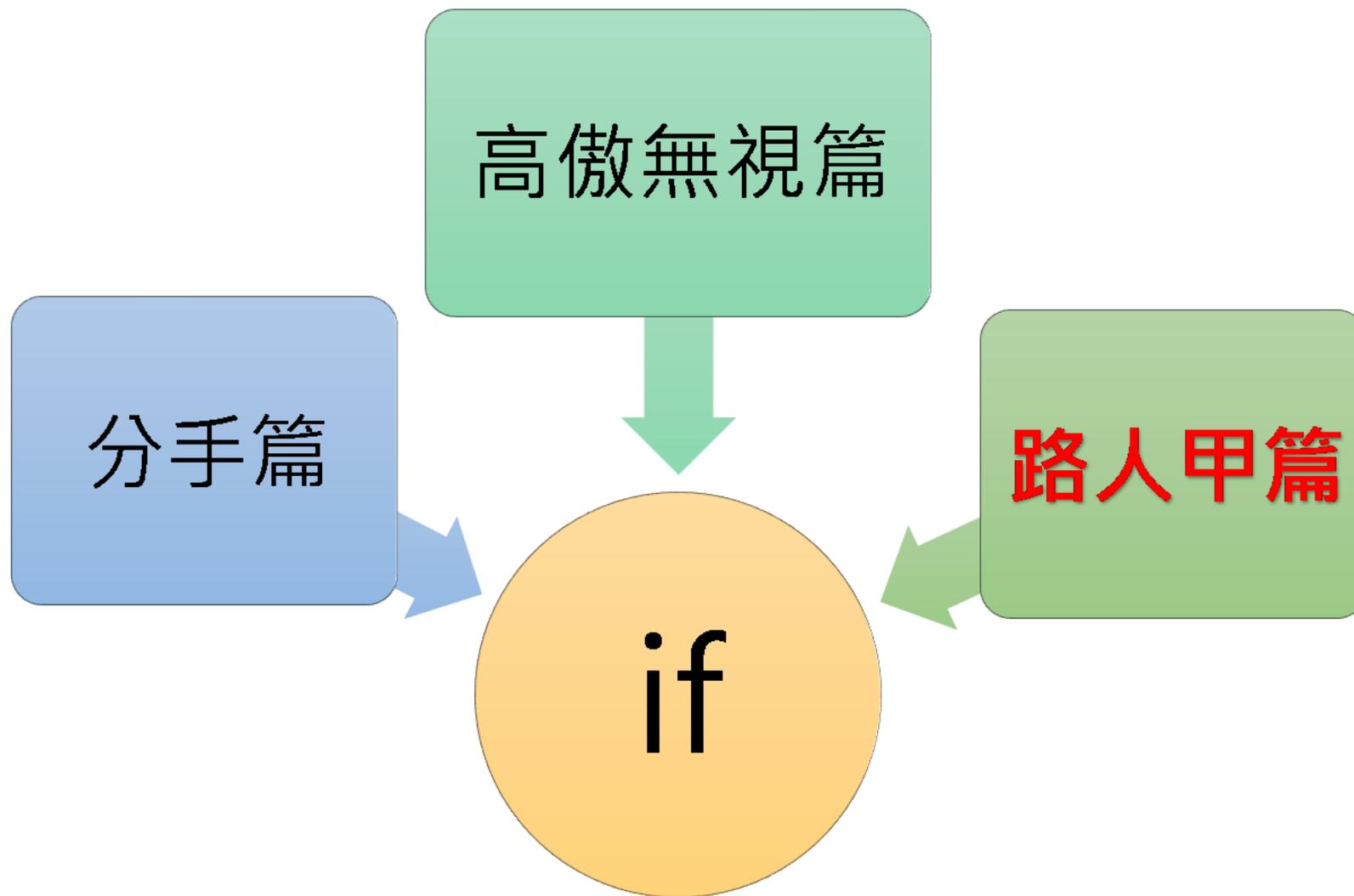
若這一回合看到不如意的，就直接跳下一回合！

遇到 continue 後，就不會繼續執行 while 迴圈區塊內 continue 後面的程式，會直接跳到 while 條件式那行，重新判斷條件式與進入 while 迴圈

執行結果

```
6
12
18
30
```

if 小劇場很搶戲 – 路人甲篇



if 小劇場很搶戲 – 路人甲篇 pass

```
i = 1
while i <= 30:
    if i % 8 == 0:
        pass

    if i % 6 == 0:
        print(i)
    i = i + 1
```

就只是路過看一下，
沒有任何動作的！

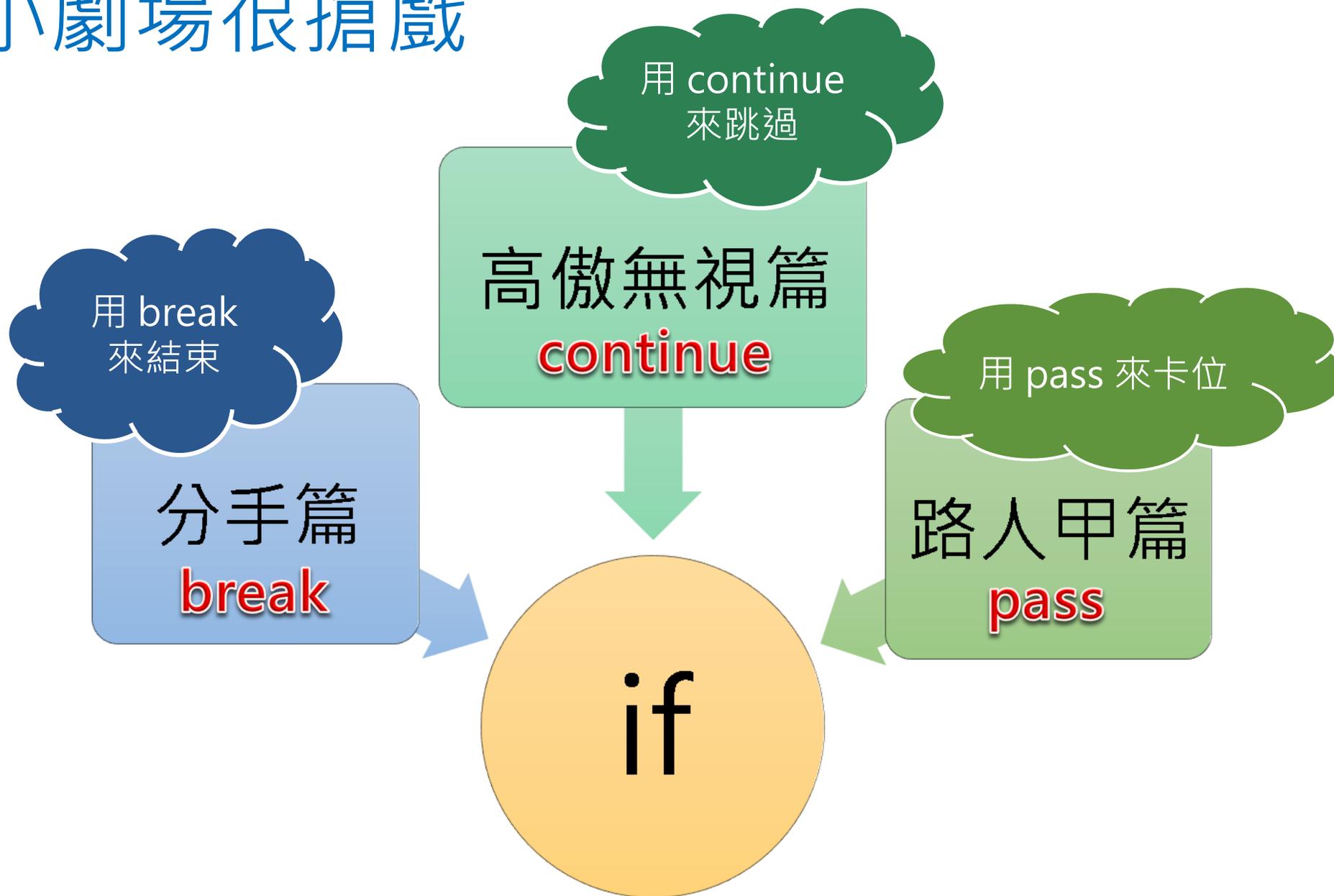
pass 不會執行任何動作，大
多時後只是透過 pass 預留空
間。

執行結果

```
6
12
18
24
30
```

以此例來講，也許之後會調整成
`if i % 8 == 0:`
`print('{num}是8的倍數'.format(num=i))`
但在還不需要印出訊息前，就先用
pass 保留那個位置

if 小劇場很搶戲



整理一下

- **break**: 結束整個迴圈
- **continue**: 跳過該次循環，進入迴圈的下個循環
- **pass**: 先卡位，不執行，保留位置並維持結構完整性

PYTHON

延伸的概念

概念 1: 小心跑不完

在執行 continue 之前，要確認下一回合不會重複這一回合的執行情形，所以與條件式判斷有關的變數值一定要有更動才能進入下一回合

```
i = 1
while i <= 30:
    i = i + 1
    if i % 8 == 0:
        continue
    if i % 6 == 0:
        print(i)
```

```
i = 1
while i <= 30:
    if i % 8 == 0:
        continue
    if i % 6 == 0:
        print(i)
    i = i + 1
```

小心！放在這裡會造成無限迴圈，while 永遠跳不出去！

概念 2: while... else...

- 當循環有正常結束時，會執行後面的 else 區塊
- 若是透過 break 不正常結束的循環，就不會執行 else 區塊
- 可以用來檢查迴圈是否有被中斷

```
i = 0
while i < 10:
    i = i + 1
else:
    print('我有從1跑到10囉')
```



```
i = 0
while i < 10:
    if i > 6:
        break
    i = i + 1
else:
    print('我有從1跑到10囉')
```

