


程式設計概論

Programming 101

—程式迴圈

授課老師：邱淑怡

DATE: 10/16/2023



Outline

- 迴圈結構

- ◆ for 迴圈

- for迴圈稱之為迭代器迴圈(計數器迴圈)，迴圈次數取決於資料包含元素的個數

- ◆ while 迴圈

- while迴圈稱為條件迴圈，迴圈次數取決於條件何時布林值為False

敘述(statement)

- range() function
- for loop
- while loop
- break
- continue
- pass

range() function

□ range(start, stop[, step]) ，傳入的參數必須是(正、負)整數

- start為起始值(含)，預設值從0開始
- stop為結束值(不含)
- step為遞增值，預設值為1

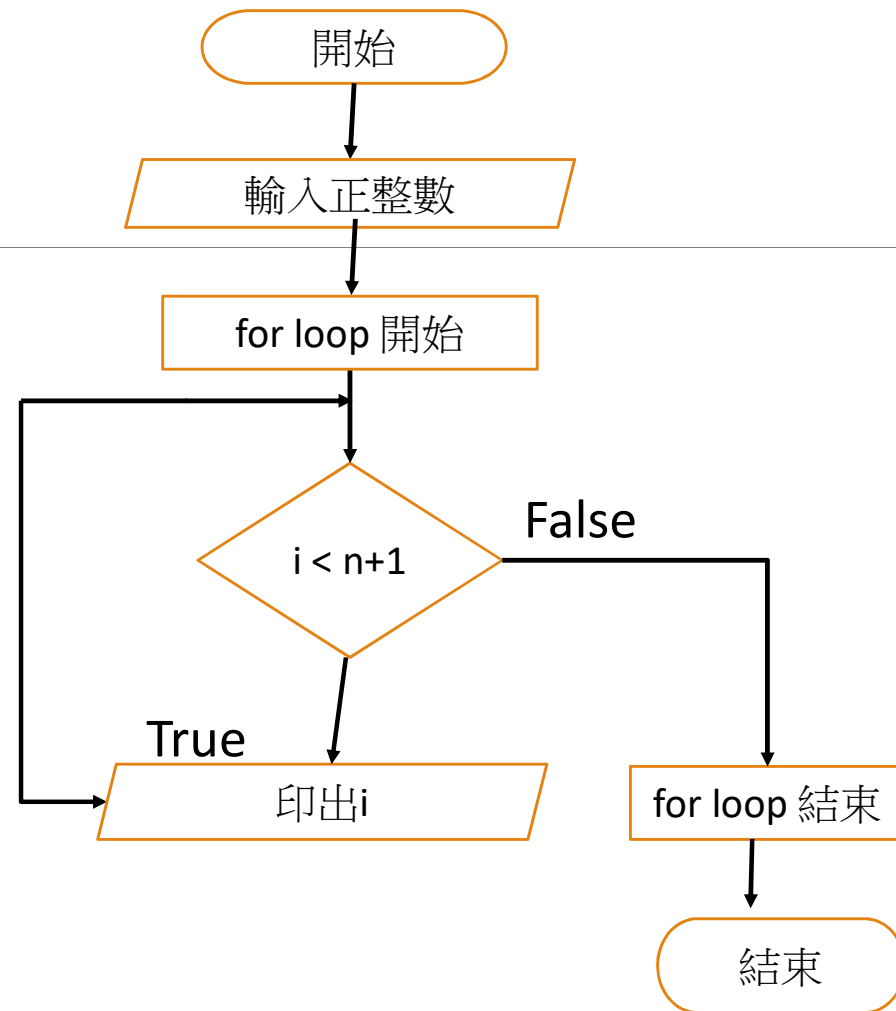
字串(string)及串列(list)亦是相同概念

```
r=range(5)
print(r)
print(list(r))
```

```
print(list(range(10))) # range(0,10)
print(list(range(0,15,5))) #step 5, [0,5,10]
print(list(range(0,10,3))) # [0, 3, 6, 9]
print(list(range(0,-10,-2))) # [0, -2, -4, -6, -8]
```

for loop 結構

for loop 流程圖



for 結構語法

使用控制變數來控制
for迴圈的執行次數

- 迴圈(loop)中的程式敘述，會依據**控制變數**數值的變化，被重複執行
- for迴圈又稱為”計數迴圈”，而此控制變數則稱為”計數器”
- for loop語法:

for **變數** in 串列或任何一串數字或文字:

 程式敘述 # 可多行程式敘述

```
n=int(input("輸入正整數:"))
for i in range(1,n+1):
    print(i,end=' ')
print("完成for迴圈的工作了")
```

range(a,b) means (a,a+1,...,b-1)

Use <http://pythontutor.com/>

重複結構語法

該變數可視為索引
值 (index)或計數器

- 迴圈(loop)中的程式敘述，會依據**控制變數**數值的變化，被重複執行
- for **變數** in 串列或任何一串數字或文字:
- statement3 # 可多行程式敘述
- for **index1** in range(1,10):
- print(index1)
- print(“完成for迴圈的工作了”)

Python range() function語法:

range(start, stop[, step])

Ex:

list(range(10)) # range(0,10)

list(range(0,15,5)) #步長 5, [0,5,10,15]

list(range(0,10,3)) # [0, 3, 6, 9]

list(range(0,-10,-2)) # [0, -2, -4, -6, -8, -10]

for 結構語法(cont.)

請撰寫一個Python程式，
它可以從1加到5計算總
和？

```
Result=0
for i in range(1,6):
    Result=Result+i # Result+= i
    print(Result, i)
print("完成for迴圈的工作了, Result:", Result)
```

迴圈次數	=左邊的結果	i	=右邊的結果
第1次	0	1	0+1
第2次	1	2	1+2 (3)
第3次	3	3	3+3 (6)
第4次	6	4	6+4 (10)
第5次	10	5	10+5 (15)

練習題

- 撰寫一個Python程式，令它找出1-100之間可以被13整除的數字，並印出結果

for 重複結構 – 重複執行某個動作

撰寫Python程式

- Q1: 請計算 $1+2+3+\dots+100$ ，印出總和
- Q2: 請計算 $1\sim 100$ ，印出偶數的總和
- Q3: 請找出 $1-100$ 之間可以被13整除的數字，並印出結果

for重複結構 – 重複執行某個動作

□ 練習題

- 若要計算 $1+2+3+\dots+100$ ，印出總和

```
Result=0
for i in range(1,101):
    Result+= i #Result=Result+i
print(f"完成for迴圈的工作相加結果是:{Result}")
```

重複結構語法

該變數可視為索引
值 (index)或計數器



□ 迴圈(loop)中的程式敘述，會依據**變數**數值的變化，被重複執行

for **變數(identifier)** in 串列或任何一串數字或字串:

```
statement1 # 可多行程式敘述
```

```
statement2
```

```
for i in range(1,10):
```

```
    print(i)
```

```
print("完成for迴圈的工作了")
```

range() function 可以指定迴圈中的 index (i) 的變化

```
for i in range(1,20,3):    #i is the index
    print(i)
print("完成for迴圈的工作1了")
```

```
for i in range(20,1,-2):
    print(i)
print("完成for迴圈的工作2了")
```

用list(串列)資料型態與for結合

- 串列(list) 中每個元素皆印出其值及對應的索引值
 - `range(len(a)) = range(6) = range(0:6) == range`函式會依序傳回0,1,2,3,4,**5(6-1)**

```
a= [3,6,-7,1,-4,12]
for i in range(len(a)):
    print(i, a[i])
print("完成串列迴圈工作")
```

多個for迴圈

```
for i in range(a,b):  
    for j in range(c,d):  
        print(i,j)  
        print("內迴圈")  
    print("外迴圈")  
print("所有迴圈都結束了")
```

c, d 數值可能會與i 有關

多個for迴圈_實例說明

```
for i in range(2,6):  
    for j in range(1,i):  
        print(i,j)  
        print("內迴圈")  
    print("外迴圈")  
print("所有迴圈都結束了")
```

多個for迴圈_實例

□九九乘法表

```
for i in range(1,10):  
    for j in range(1,10):  
        s= i*j  
        print('%d * %d = %d ' %(i, j , s))
```

```
for i in range(1,10):  
    for j in range(1,10):  
        s= i*j  
        print('%d * %d = %d ' %(i, j , s), end="")  
    print('\n')
```

end=' '意思
是末尾不換
行，加空格

多個for迴圈_實例(cont.)

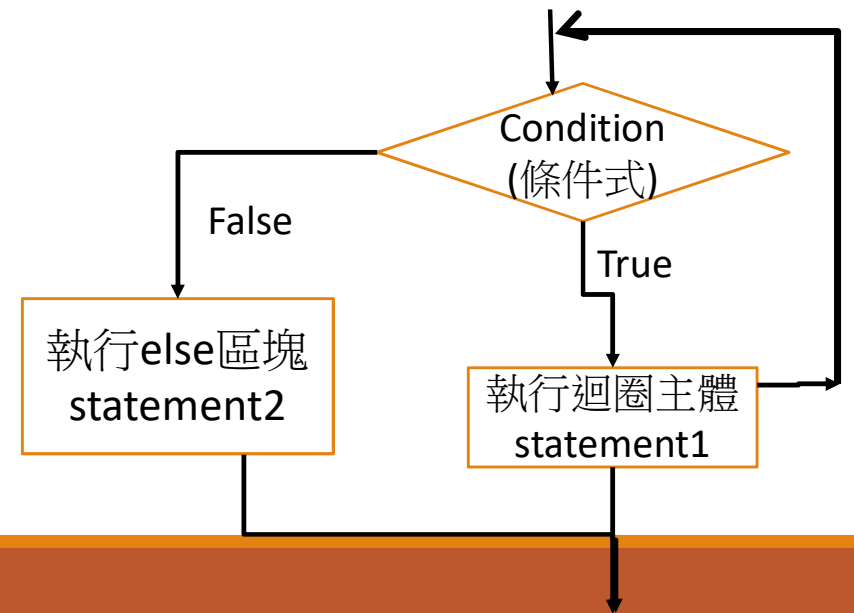
□九九乘法表

```
k=[1,2,3,4,5,6,7,8,9] # k is a list(串列)
for i in k:
    for j in k:
        s=i*j
        print('%d * %d = %d' %(i, j, s), end=" ")
    print('\n') # 換行
```

while迴圈

- 迴圈是以條件式是否成立做為是否執行迴圈的依據，稱條件式迴圈
- 在while中，最重要的事情就是「condition, 條件(問題)」的答案是否有「改變」，當問題的答案由True變成False的時候，才會跳出迴圈，停止重複
- while condition(條件):
 - statement1
 - [else:
 - statement2]

Use <http://pythontutor.com/>



while迴圈_實例說明

- while迴圈避免無窮迴圈
- 用while迴圈印出0,1,2,3,4

```
i=0
while i<5:
    print(i)
    i=i+1
```

Use <http://pythontutor.com/>

while...else

```
i=0
while i<5:
    print(i)
    i=i+1
else:
    print("while else")
```

break 敘述

□ break

- break 可以讓你跳出圈圈，不管你已經跑在迴圈的哪個位置，它絕對會讓你「立刻跳出迴圈」，執行下一個程式區塊。

```
while True:
    data = input('若要結束程式請輸入stop:')
    if data.lower()=='stop':
        break
    print(data)
print('Finished')
```

continue 敘述

□ continue

- continue 不會讓你跳出迴圈，它跟break一樣都是打斷你的迴圈，但是continue會再帶你「回到迴圈的起點」，繼續繞下一次的迴圈。

```
while True:
```

```
    data = input('若要留在迴圈內輸入cont, 若要結束程式請輸入stop: ')
```

```
    if data == 'stop':
```

```
        break
```

```
    if data == 'cont':
```

```
        print('在continue 內')
```

```
        continue
```

```
    print(data)
```

```
print("OK, stop it.")
```


pass 敘述

- 空語法，為了保持程序結構的完整性
- Example

```
for var in 'Python code':  
    if var=='':  
        pass  
    else:  
        print(var)
```

多個while迴圈_實例

□九九乘法表

```
i=1 # 需要給i,j 起始值
while i < 10:
    j=1
    while j < 10:
        s=i*j
        print ('%d * %d = %d' %(i, j, s))
        j=j+1    # j +=1
    i=i+1 # i += 1
    print('\n')
```

練習題

練習題1

□[百貨公司折扣戰]

- 消費金額10萬元以上就打七折，若金額在5萬元以上就打八折，若金額在3萬元以上就打九折，若金額在1萬元以上就打九五折
- 請撰寫一個Python程式，幫該公司設計收銀台的程式，輸入顧客消費金額後，判斷顧客應付的金額。

練習題2：輸入三個邊長，判斷能否組成一個三角形？

- 讓使用者輸入三角形的三個邊長，若無法組成三角形，則讓使用者再次輸入三個邊長，直到能形成三角形後計算該三角形的周長，並結束輸入的工作
- 組成三角形的條件：
 - 三角形任二邊長和大於第三邊
 - 三角形任二邊長差小於第三邊

練習題3

Print the following pattern on the screen using for loop

```
#
```

```
 #
```

```
  #
```

```
   #
```

```
    #
```

```
   #
```

```
  #
```

```
 #
```

```
#
```

練習題4

- 請加總 $-1+(-3)+(-5)+\dots+(-99)$
- (hint: 可練習 `range()` function)

練習題5:延續“猜數字的小程式”完整版

- 程式設計人員由電腦任意產生一個兩位數以內的正整數當作真值，接著讓使用者猜這個數字(“請使用者輸入一個兩位數字以內的正整數:”)。
- 接著，程式進行比較大小，若猜測數字 $>$ 真值，請印出“你猜的數字太大”；若猜測數字 $<$ 真值，請印出“你猜的數字太小”；若剛好相等，請印出“恭喜你猜對了”。當使用者沒猜對就讓使用者繼續猜下一個正整數，直到猜對為止才離開程式。
- 可以在進一步縮小範圍讓使用者最快能猜到數字

Python Projects Junior Developers Can Build for Coding Practice

<https://www.freecodecamp.org/news/python-projects-junior-developers/>

補充題

練習題6: 猜數字1A2B

1. 數字不能重複
2. 限數字1,2,3,4,5,6,7,8,9
3. 根據上面兩個規定隨機產生四個數字，如: 9312
4. 使用者根據線索來猜這個數字是多少

1. 數字不重複
2. 限數字1,2,3,4,5,6,7,8,9
3. 根據上面兩規則隨機產生四位數字，例如: 1357, 9312
4. 使用者根據線索來猜這個數字是多少

