

## ▶ Ch04 基礎程式設計實作



### 選擇題

#### 😊 實力挑戰

- ( B ) 1. 有兩個陣列 $a=[3,6,5]$ ， $b=[2,6,8]$ ， $c=a[0]+b[2]$ ，請問 $c$ 的值為何？  
(A)2 (B)11 (C)9 (D)12。
- ( A ) 2. 有一個二維陣列 $x=[[1,2,3],[4,5,6],[7,8,9]]$ ，在Python語言中，若我們要取得6，要透過以下哪種方式？(A) $x[1][2]$  (B) $x[0][3]$  (C) $x[2][3]$  (D) $x[1][1]$ 。
- ( C ) 3. 我們會使用\_\_\_\_\_ (參數1, 參數2, ....)來呼叫定義好的函數，其中，空白處應該填入？(A)函數的參數 (B)函數的引數 (C)函數的名稱 (D)函數的回傳值。
- ( D ) 4. 如果想要將原本長達100行的重複程式碼簡化，可以嘗試使用下列何種結構？(A)宣告結構 (B)循序結構 (C)選擇結構 (D)重複結構。
- ( C ) 5. 執行以下的程式碼後，螢幕上顯示的數值為何？(A)37 (B)25 (C)50 (D)42。

解析：

```
1 def recursive_add(n):  
2     if n <= 2:  
3         return 5  
4     return n + recursive_add(n - 3)  
5 print(recursive_add(15))
```

5.  $15 + 12 + 9 + 6 + 3 + 5 = 50$   
(答案在 Ch4E-s5.py)

#### 😊 實力關卡

(第 1~6 題，請依下列情境描述作答)

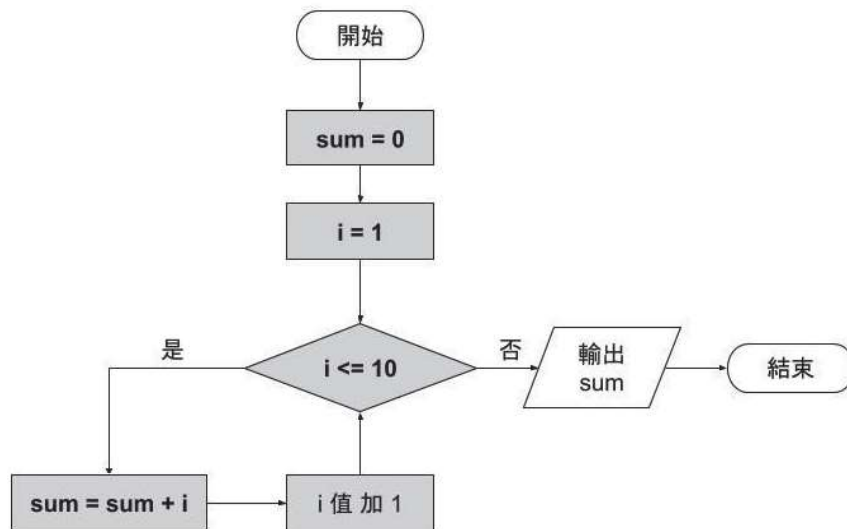
Cathy 報名參加校際盃「小程式大考驗」，其中一題只給流程圖及對應的Python 程式，如下：

(提示：range(1,11) 代表 [1,2,3,...,10]; 以下為待修正的 Python 程式碼)

```

1 sum = 0
2 i = 1
3 for i in range(1,11)
4 sum = sum + i
5 print(sum)

```



- (C) 1. 第3行程式碼有明顯語法錯誤，請問應如何修改？(A)應改為「for i in [1,11]」 (B)應改為「for i in (1,11)」 (C)應改為「for i in range(1,11):」 (D)應改為「for i in range(1,10):」。
- (B) 2. 第4行程式碼出現語意錯誤，請問是怎樣的錯誤？(A)sum 不可能等於 sum + 1」 (B)應向右縮排 (C)最右結束處應加冒號「:」 (D)應改為「sum == sum + i」。
- (B) 3. 修正第3及4行的錯誤後，Python程式碼執行到最後時，「i」應為何？(A)1 (B)10 (C)11 (D)55。
- (C) 4. 修正第3及4行的錯誤後，Python程式碼執行到最後時，「sum」應為何？(A)0 (B)45 (C)55 (D)66。
- (C) 5. 若依照此程式流程圖，最後「i」應為何？(A)1 (B)10 (C)11 (D)55。

(C) 6. 若依照此程式流程圖，最後「sum」應為何？(A)0 (B)45 (C)55 (D)66。

(B) 7. 下列Python程式片段，執行之後的顯示結果為何？(A)5 (B)10 (C)50 (D)2。

```
a=5
b=10
if a < b :
    a = b
print(a)
```

(C) 8. 使用Python程式語言執行下列程式碼後，則螢幕輸出結果是多少？(A)0.5 (B)1.0 (C)2.0 (D)4.0。

```
a=8
b=4
c=2
if (a / b) > c:
    print(b / a)
else:
    print(b / c)
```

(C) 9. 執行下列Python程式片段後，則螢幕輸出結果是多少？(A)1000 (B)1369 (C)631 (D)369。

```
sum = 1000
for i in [3,33,333]:
    sum = sum + i
print(sum)
```

(C) 10. 下列何者不是結構化程式設計的控制結構之一？(A)循序結構 (B)選擇結構 (C)宣告結構 (D)重複結構。

(A) 11. 結構化的程式語言應避免採用下列哪一種結構？(A)跳躍結構 (GOTO) (B)重複結構 (C)選擇結構 (D)循序結構。

(A) 12. 衛生福利部根據相關研究，公布臺灣成人肥胖標準：BMI < 18.5 為過輕， $18.5 \leq \text{BMI} < 24$  為正常體重，否則就算過重， $\text{BMI} = \frac{\text{體重(公斤)}}{\text{身高}^2(\text{公尺}^2)}$ 。以下Python程式碼中，兩個空格處①及②，應分別填多少？(A)18.5、24 (B)18.5、18.5 (C)24、18.5 (D)24、24。

```

x = 1.60      # 單位：公尺
y = 50       # 單位：公斤
BMI = y / (x*x)
print(BMI)
if BMI < ① :
    print('體重過輕！')
elif BMI < ② :
    print('體重合適！')
else:
    print('體重過重！')

```

## 多元練習

1. 嘉明參加鐵人三項競賽，奪得冠軍，他的比賽成績如下表，

游泳	單車	跑步
30分鐘	1小時	1小時半

請利用Python程式將嘉明的「比賽項目名稱：成績」，顯示在電腦螢幕上，最終成果如下圖。

```

游泳 : 30分鐘
單車 : 1小時
跑步 : 1小時半

```

2. 假設我們有三位學生的國文、數學與自然成績（如下表），

	國文	數學	自然
嘉明	100	20	85
小美	95	99	75
阿雄	89	73	92

現在請利用Python程式雙層for迴圈，計算每個人的總成績（三個科目的總和），並顯示在電腦螢幕上（如下圖）。

```
嘉明 總分為 205 分  
小美 總分為 269 分  
阿雄 總分為 254 分
```

3. 請利用Python寫一支程式，統計下列成績中，

成績	40	80	75	20	96	69	50
----	----	----	----	----	----	----	----

不及格（低於60分）以及成績高於90分的人分別有多少，並將統計結果顯示在電腦螢幕上（如下圖）。

```
<60分: 3 個  
>90分: 1 個
```

4. 修改4-4節的實例演練程式，使得輸入為負數( $n < 0$ )時，函數factorial( )則回傳「請輸入0或正整數」。

5. 請利用Python實作一個函數，回傳輸入的兩個參數p, q的最大公因數。即：

```
1 def gcd(p, q):  
2     # 你的程式碼區塊  
3     return .....
```

提示：

- (1) 可以嘗試使用遞迴的概念
- (2)  $p \div q$ 的餘數可以用%運算符號

#### 解析

- 1.( 答案在 Ch4E-1.py)
- 2.( 答案在 Ch4E-2.py)
- 3.( 答案在 Ch4E-3.py)
- 4.( 答案在 Ch4E-4.py)
- 5.( 答案在 Ch4E-5.py)