

## CH - 5 基本程式編寫概念

### 練習卷 #3

小翠及智仁協助羅先生設計一個電腦系統，用以尋找輸入分數之對應等級。

(a) 小翠使用以下偽代碼來展示她的算法：

步驟 1：輸入一個數值，並儲存在 MARK 內

步驟 2：若  $MARK < 40$  則  $GRADE \leftarrow 'Unattained'$

步驟 3：若  $MARK \geq 40$  則  $GRADE \leftarrow 'Attained'$

步驟 4：若  $MARK \geq 80$  則  $GRADE \leftarrow 'Distinction'$

步驟 5：輸出 GRADE

(i) 試就以下每個 MARK 的數值，寫出 GRADE 內的值。

(1)  $MARK = 40$  (1 分)

(2)  $MARK = 200$  (1 分)

(ii) 小翠採用迭代控制結構來修改步驟 1 之偽代碼，使 MARK 值介乎 0 和 100 之間，包括首尾兩數，如下所示。

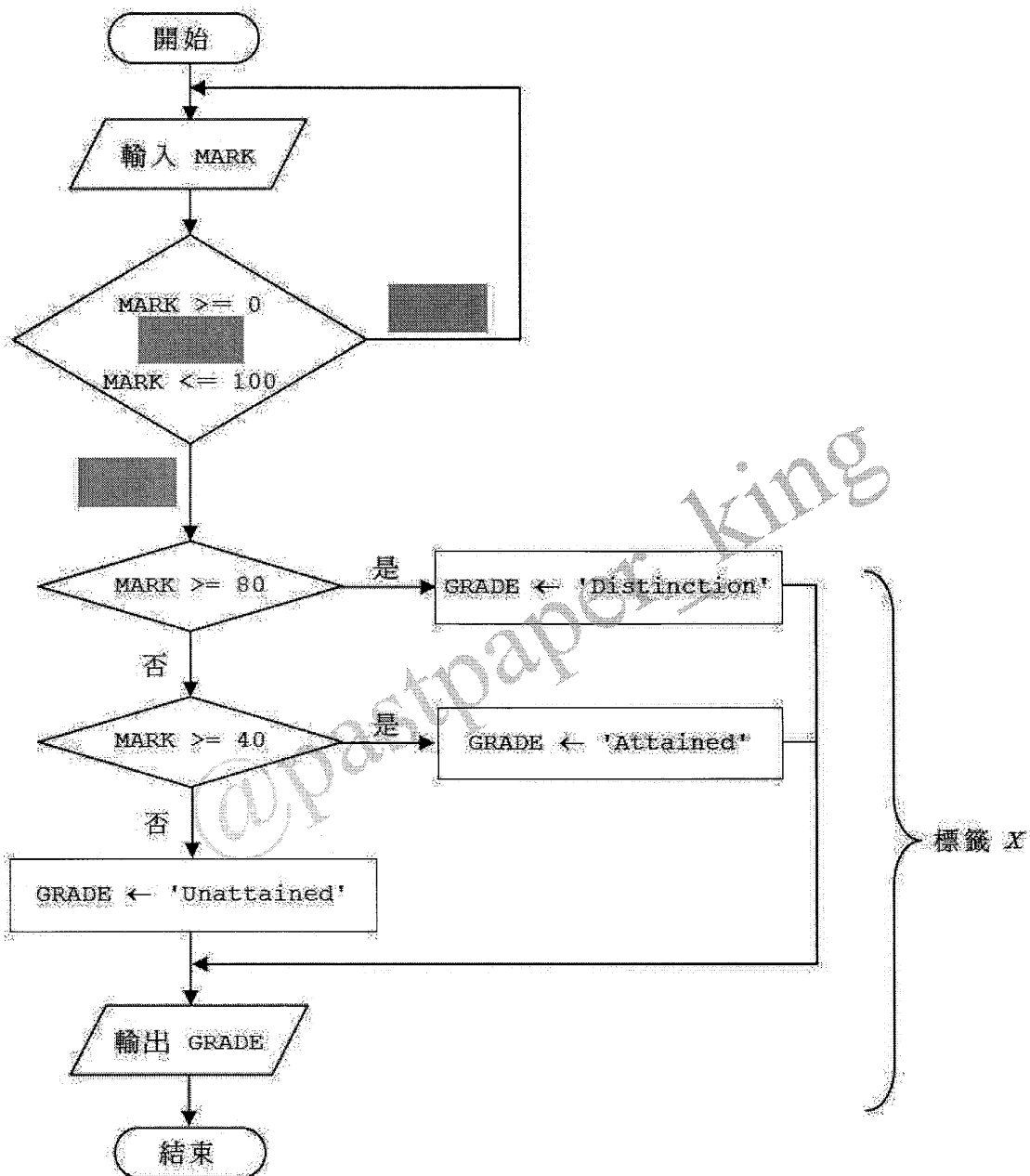
步驟 1：當 MARK 值小於 0 或大於 100，重複輸入數值至 MARK 內。

(1) 除了 0 和 100，試寫出另一個可以用來識別算法的邊際個案的測試數據。 (1 分)

(2) 小翠使用了前期測試、後期測試，抑或 for 循環類型的迭代控制結構呢？ (1 分)

練習卷 #3 | 續

(b) 智仁使用下列流程圖來展示他的算法。在第一個判定框之陰影區內，填寫「是」、「否」及適合之運算符。



(2 分)

(c) 比較標籤 X 範圍內的算法與小翠的算法，哪一個更為有效率？試簡略說明。

(2 分)

**2012 #3**

考慮以下的算法。

```
A ← 1
B ← 1
當 B < 20 執行
    A ← A + B
    B ← A - B
輸出 B
```

(a) 空運行這個算法，並寫出首三個輸出。

(2 分)

(b) (i) 該算法被重寫，使輸出 B 的次數存放在 N 內。試完成下列的算法。

```
A ← 1
B ← 1
N ← (1) _____
當 B < 20 執行
    A ← A + B
    B ← A - B
    N ← (2) _____
輸出 B
輸出 N
```

(2 分)

(ii) 寫出 B 和 N 的最終值。

(2 分)

(c) 假設 A 和 B 的初始值改變了。第一次執行「輸出 B」後，A 和 B 的值分別為 9 和 4。A 和 B 的初始值是什麼？

(2 分)

**2013 #5**

某百貨公司送給會員一張優惠咭。其中一名會員小芬在某月份使用此優惠咭進行 10 項交易，相關金額儲存在  $T[1], T[2], \dots, T[10]$ 。

(a) 完成下列算法 ALG1，以計算小芬的總金額 TOTAL。

**ALG1**

TOTAL  $\leftarrow$  (i) \_\_\_\_\_

I  $\leftarrow$  1

當 I (ii) \_\_\_\_\_ 10 執行

    TOTAL  $\leftarrow$  (iii) \_\_\_\_\_

    I  $\leftarrow$  I + 1

(3 分)

該百貨公司設有一購物獎賞推廣，每月會員使用此優惠咭進行首 10 項交易中，毋需繳付金額最低的一項交易。

(b) (i) 完成下列算法 ALG2，以計算小芬金額最低的一項交易。

**ALG2**

M  $\leftarrow$  1

J  $\leftarrow$  1

當 J  $\leq$  10 執行

    如果 T[M]   T[J] 執行

        M  $\leftarrow$  J

    J  $\leftarrow$  J + 1

(1 分)

(ii) 可否將 ALG2 的第二行改成 'J  $\leftarrow$  2'？試簡略說明。

(1 分)

(c) 在此購物獎賞推廣中，ALG3 是用來計算小芬實際應付金額 PAYMENT。ALG3 包含 ALG1 和 ALG2。試完成 ALG3。

**ALG3**

執行 (i) \_\_\_\_\_

執行 (ii) \_\_\_\_\_

PAYMENT  $\leftarrow$  (iii) \_\_\_\_\_

(3 分)

**2014 #3**

以下的算法 ALG1 和 ALG2 幫助小明找出一組數字序列。

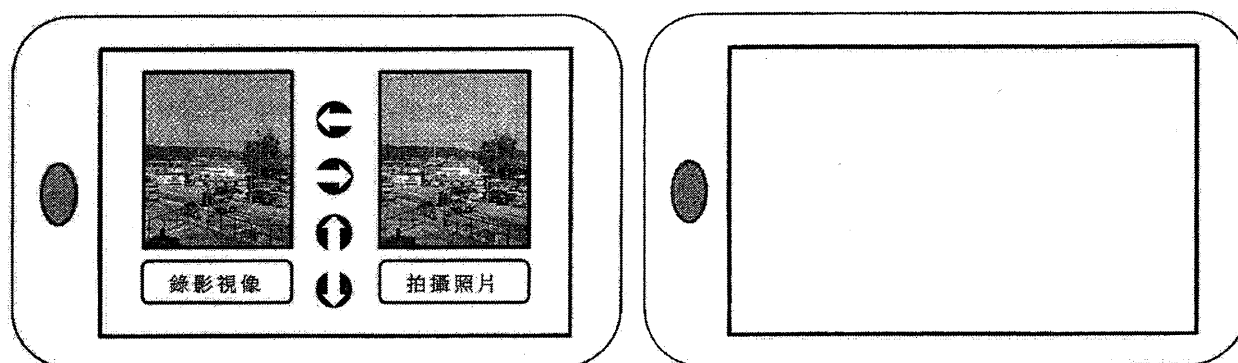
<u>ALG1</u>	<u>ALG2</u>
$A \leftarrow 0$ $B \leftarrow 1$ $P \leftarrow 10$ 當 $A \leq P$ 執行 輸出 A $TEMP \leftarrow A + B$ $A \leftarrow B$ $B \leftarrow TEMP$	$A \leftarrow 0$ $B \leftarrow 1$ $Q \leftarrow 5$ 設 I 由 1 至 Q 執行 輸出 A $TEMP \leftarrow A + B$ $A \leftarrow B$ $B \leftarrow TEMP$

- (a) (i) 寫出 ALG1 的輸出數值。 (2 分)
- (ii) 寫出 ALG2 的輸出數值。 (2 分)
- (iii) 如果 ALG2 的輸出數值和 ALG1 的輸出數值要一樣的話，Q 的初始值應改為什麼？ (1 分)

**2015 #1**

小美開發了一架無人機，它是一種遙遠控制的飛行裝置。她能使用一部流動裝置來控制這架無人機飛行，並從高處進行高空視像和照片拍攝。

- (d) 小美使用的流動裝置是一部平板電腦。她開發了一個界面來控制無人機的移動方向和拍攝視像及照片，如下展示（左圖）：



- (i) 這個界面設計有些缺點。試進行**兩項**修改以改善這個設計，並在上方（右圖）畫出。 (2 分)
- (ii) 試簡略描述 (d)(i) 內的改善之處。 (2 分)

**2015 #4**

黃小姐開發了一台內置電腦的自動販賣機。她編寫了一個程式來控制自動販賣機，銷售 30 種產品。

黃小姐以一個由 30 個元素  $P[1], P[2], \dots, P[30]$  所組成的陣列，儲存在自動販賣機內 30 種產品餘下的數量。她利用以下算法 ALG1 檢查自動販賣機內所有產品是否售罄。

**ALG1**

$N \leftarrow 0$

設  $I$  由 1 至 30 執行

$N \leftarrow N + P[I]$

如果  $N = 0$

則  $FLAG \leftarrow FALSE$  (FALSE 代表「假」)

否則  $FLAG \leftarrow TRUE$  (TRUE 代表「真」)

- (b) (i) 如果  $P[1]$  和  $P[2]$  的值分別是 0 和 1，FLAG 的最終值是什麼？ (1 分)
- (ii)  $N$  的最終值代表什麼意思？ (2 分)
- (iii) 黃小姐選用布爾為 FLAG 的數據類型。她的選擇有什麼主要好處？ (1 分)
- (c) (i) 根據  $I$  和 FLAG 的值，完成以下真值表。 (3 分)

$I$	FLAG	$(I \leq 30) \text{ AND } (\text{NOT FLAG})$
1	FALSE	TRUE
15	TRUE	(1)
30	FALSE	(2)
31	TRUE	(3)

- (ii) 以下算法 ALG2 達成 ALG1 相同的任務。

**ALG2**

$FLAG \leftarrow FALSE$

$I \leftarrow 1$

當  $(I \leq 30)$  與  $(\text{NOT FLAG})$  執行

如果  $P[I] > 0$

則  $FLAG \leftarrow TRUE$

$I \leftarrow I + 1$

- 黃小姐應該選用 ALG1 還是 ALG2 的算法？試簡略說明。 (1 分)

**2016 #2**

永強每天會在公司透過某電郵伺服器接收很多電子郵件。此電郵伺服器能配對某數據檔內的字串來辨別濫發郵件。陣列 ST 是儲存此數據檔內的字串，如下展示：

ST[1]	ST[2]	...	ST[1,000]	...
"恭喜"	"中獎"	...	"得獎者"	...

以下兩個算法 ALG1 和 ALG2 檢查變量 check\_ST 所儲存的字串，是否包含在此數據檔內。它們得出的結果是相同的。N 是儲存數據檔內字串總數量的變量。

<b>ALG1</b>	<b>ALG2</b>
輸入 check_ST FOUND $\leftarrow$ -1 設 I 由 1 至 N 執行 如果 check_ST = ST[I] 則 FOUND $\leftarrow$ I	輸入 check_ST FOUND $\leftarrow$ -1 I $\leftarrow$ 1 當 (I $\leq$ N) 與 (FOUND = -1) 執行 如果 check_ST = ST[I] 則 FOUND $\leftarrow$ I I $\leftarrow$ I + 1

- (d) (i) 假設在 check\_ST 內的字串是「中獎」。執行這些算法後，FOUND 的值是什麼？ (1 分)
- (ii) 試舉出兩個 check\_ST 的測試值來有效地驗證這些算法，並簡略描述你的答案。 (2 分)
- (iii) 就 (d)(i) 的個案而言，ALG1 還是 ALG2 較有效果？試簡略說明。 (2 分)
- (iv) 使用陣列儲存這些字串有什麼好處？ (1 分)

**2017 #3**

某鐵路公司開發一個電腦系統，在車站設置具有觸式顯示屏幕的資訊亭。

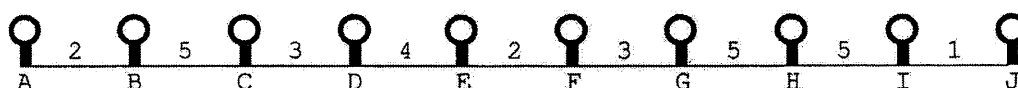
- (a) 資訊亭的顯示屏幕上應顯示以下列車的資訊，並協助乘客查找列車的到達時間和月台編號。

- 列車編號，如 3338 和 0910
- 目的地，如深圳和天津
- 到達時間，如 14:30 和 08:00
- 月台編號，如 3 和 12

試設計資訊亭的互動式版面，並加注釋。

(3 分)

在某一條路線上，從車站 A 到車站 J 共有 10 個車站，任何兩個相鄰車站之間均有一個估計的行車時間（以小時計），如下所示。舉例來說，從車站 A 到車站 B 的行車時間估計為 2 小時。



- (c) 從車站 D 到車站 G 的行車時間估計是多少？ (1 分)

### 2017 #3 | 續

在此電腦系統中，使用了陣列  $W$  中的  $W[1], W[2], \dots, W[9]$  來分別儲存 9 個估計的行車時間，即  $W[1] = 2, W[2] = 5$ ，如此類推。

(d) 完成以下算法，以計算從車站 A 到車站 J 估計的行車時間。

$X \leftarrow$  (i) \_\_\_\_\_

$Y \leftarrow$  (ii) \_\_\_\_\_

$T \leftarrow 0$

設  $i$  由  $X$  至  $Y$  執行

$T \leftarrow$  (iii) \_\_\_\_\_ + (iv) \_\_\_\_\_

輸出  $T$

(4 分)

(e) 在循環內使用變量  $X$  和  $Y$  而不使用常數的主要優點是什麼？

(2 分)

### 2018 #5

莉莉設計了一個猜數字遊戲的算法如下：

$N \leftarrow 100$

$ANSWER \leftarrow$  (一個由 1 至  $N$  的隨機整數，包含首尾兩數)

重複

輸入  $GUESS$  (整數)

如果  $(GUESS < 1)$  或  $(GUESS > N)$

則 輸出 “超出範圍！”

否則 如果  $(GUESS <> ANSWER)$

則 輸出 “再次嘗試！”

直至  $(GUESS = ANSWER)$

輸出 “你勝出了！”

(a) 假設  $ANSWER$  的值是 19。為每個輸出建議一個輸入值來測試這個算法。

(3 分)

輸入	輸出
(i)	超出範圍！
(ii)	再次嘗試！
(iii)	你勝出了！



**2018 #5 | 續**

(b) 莉莉打算在此算法進行兩項修改：

1. 如果 GUESS 是小於 ANSWER，顯示“太小！”，及  
如果 GUESS 是大於 ANSWER，顯示“太大！”；
2. 容許用戶猜最多 5 次。

(i) 完成下列算法以展示這些修改。

(4 分)

行號	內容
1	$K \leftarrow 0$
2	$N \leftarrow 100$
3	ANSWER $\leftarrow$ (一個由 1 至 N 的隨機整數，包含首尾兩數)
4	重複
5	輸入 GUESS (整數)
6	如果 (GUESS < 1) 或 (GUESS > N)
7	則 輸出“超出範圍！”
8	否則 如果 ((1) _____)
9	則 輸出“太小！”
10	否則 如果 ((2) _____)
11	則 輸出“太大！”
12	$K \leftarrow$ (3) _____
13	直至 (GUESS = ANSWER) 或 ((4) _____)
14	如果 (GUESS = ANSWER)
15	則 輸出“你勝出了！”

(ii) 如果刪除行號 14「如果 (GUESS = ANSWER)」，這個算法的輸出可能是錯誤的。為什麼？

(1 分)

**2019 #5**

志明設計了一個程式，使用附有陣列 A 的算法，如下展示：

$N \leftarrow 6$   
 設 I 由 1 至 N 執行  
      $A[I] \leftarrow 1 - A[I]$

(a) (i) 假設 A 的初始內容是：

A[6]	A[5]	A[4]	A[3]	A[2]	A[1]
0	0	1	1	0	0

執行此算法後 A 的內容是什麼？

(2 分)

(ii) 假設執行此算法後 A 的內容是：

A[6]	A[5]	A[4]	A[3]	A[2]	A[1]
1	0	1	0	1	0

A 的初始內容是什麼？

(1 分)

**2019 #5 | 續**

志明修改了算法如下：

 $N \leftarrow 6$  $K \leftarrow 2$ 設  $I$  由 1 至  $N$  執行如果  $K = 1$ 則  $A[I] \leftarrow 1 - A[I]$ 如果  $A[I] = 1$ 則  $K \leftarrow 1$ (b) (i) 假設  $A$  的初始內容是：

$A[6]$	$A[5]$	$A[4]$	$A[3]$	$A[2]$	$A[1]$
0	0	1	1	0	0

(1) 執行此算法一次後  $A$  的內容是什麼？

(2 分)

(2) 再執行此算法一次後  $A$  的內容是什麼？

(1 分)

(ii) 假設執行此算法後  $A$  的內容是：

$A[6]$	$A[5]$	$A[4]$	$A[3]$	$A[2]$	$A[1]$
1	0	1	0	1	0

 $A$  的初始內容是什麼？

(2 分)

$A[6]$	$A[5]$	$A[4]$	$A[3]$	$A[2]$	$A[1]$
0	1	0			

(iii) 寫出  $A$  的初始內容，使得執行此算法後，其內容仍保持不變。

(1 分)

$A[6]$	$A[5]$	$A[4]$	$A[3]$	$A[2]$	$A[1]$
0	0	0			

(c) 志明將會在他的桌上電腦以  $N \geq 64$  執行該程式數百萬次以上，並希望大幅改善此程式運行的表現。

(i) 增加 RAM 的大小是沒有幫助的。為什麼？

(1 分)

(ii) 建議一個需要升級的硬件部件。

(1 分)

**2020 #2**

小芬編寫一個涉及整數的算法，如下所示：

行號	內容
10	輸入 A, B
20	重複
30	重複 如果 $A < B$
40	則 $T \leftarrow A$
50	(i) _____ $\leftarrow$ (ii) _____
60	(iii) _____ $\leftarrow$ (iv) _____
70	$A \leftarrow ((A/B) \text{ 的餘數})$
80	直至 ( $A = 0$ 或 $B = 0$ )
90	輸出 ( $A + B$ )

(b) 第 40 至 60 行是用作交換 A 和 B 的內容。T 是一個暫存變量。完成上面的偽代碼。 (2 分)

(c) (i) 假設 A 和 B 的輸入值分別為 9 和 15。每次執行第 80 行後，寫下 A 和 B 的內容。 (3 分)

迭代	A	B
第一次	(1)	(2)
第二次	(3)	(4)
第三次	(5)	(6)

(ii) 假設 A 和 B 的輸入值是相同的正整數。第 80 行將會被執行多少次？ (1 分)

(d) 小芬在第 10 行之後加入以下偽代碼，以確保 A 和 B 的初始值為正數。完成此偽代碼。

當 ( $A$  (i) \_\_\_\_\_ 0) (ii) \_\_\_\_\_ ( $B$  (iii) \_\_\_\_\_ 0) 執行

輸出 “輸入無效！”

輸入 A, B

(2 分)

(e) 小芬使用「如果-則-否則」語句重寫第 30 至 70 行，如下所示。完成以下的語句。

如果  $A < B$

則  $B \leftarrow$  (i) \_\_\_\_\_

否則  $A \leftarrow$  (ii) \_\_\_\_\_

(2 分)

2020 #4

志明計劃在他的快餐店裏安裝一些自助點餐機，使顧客可自行點菜。然而，顧客仍可以透過收銀員使用專用終端機點菜。

(b) 以下是從點餐機的觸式屏幕所擷取的兩個畫面樣本：

飯餐	雪糕	⊖ 0 ⊕
飲品	蘋果批	⊖ 1 ⊕
	芝士蛋糕	⊖ 0 ⊕
甜點	鮮果撻	⊖ 3 ⊕
	曲奇	⊖ 0 ⊕
小吃	甜甜圈	⊖ 0 ⊕
結帳		

飯餐	芝士漢堡包	\$30 × 4
飲品	冰檸檬茶	\$20 × 4
	蘋果批	\$10 × 1
甜點	鮮果撻	\$10 × 3
	炸薯條	\$20 × 4
小吃	總額 =	\$320
結帳	確定？	<input type="button" value="是"/> <input type="button" value="否"/>

(i) 描述使用 ⊖ 和 ⊕ 按鈕的兩個優點。

(2 分)

(ii) 菜單中將添加二十個不同的甜點項目。草擬一個方便顧客使用的設計以包括新的甜點項目，並簡略描述你的設計。

(2 分)

飯餐	
飲品	
甜點	
小吃	
結帳	

**2021 #3**

志明設計了一個算法來加密二進制數字的陣列 A，如下所示：

$I \leftarrow 5$

重複

$I \leftarrow I - 1$

$A[I] \leftarrow 1 - A[I]$

直至  $A[I] = 1$

- (a) (i) 假設 A 的初始內容是：

A[1]	A[2]	A[3]	A[4]
0	0	1	1

執行此算法後 A 的內容是什麼？

(2 分)

- (ii) 假設執行此算法後 A 的內容是：

A[1]	A[2]	A[3]	A[4]
1	0	1	0

A 的初始內容是什麼？

(2 分)

- (iii) 如果 A 的初始內容中的所有數值均為 1，則該算法無法正常終止。修改該算法，使其可以正常終止。

$I \leftarrow 5$

重複

$I \leftarrow I - 1$

$A[I] \leftarrow 1 - A[I]$

直至  $A[I] = 1$  (1) \_\_\_\_\_ (2) \_\_\_\_\_

(2 分)

- (b) 志明重寫 (a)(iii) 內的算法為另一個可行算法。完成以下算法。

FLAG  $\leftarrow 5$

設 I 由 (i) \_\_\_\_\_ 遞減至 1 執行

如果 FLAG = (ii) \_\_\_\_\_ 則

$A[I] \leftarrow 1 - A[I]$

如果  $A[I] =$  (iii) \_\_\_\_\_ 則

FLAG  $\leftarrow 0$

(3 分)

## 2022 #5

莉莉設計附有陣列 A 的子程式。她考慮三個算法 ALG1、ALG2 和 ALG3。

ALG1

行號	內容
1	$B \leftarrow 0$
2	設 I 由 1 至 N 執行
3	如果 $A[I] = 1$ 則
4	$B \leftarrow B + 1$
5	如果 $B \geq N / 2$
6	則 $CH \leftarrow 'X'$
7	否則 $CH \leftarrow 'Y'$

(a) 假設  $N = 6$ 。以下列 A 的初始內容執行 ALG1。

A[1]	A[2]	A[3]	A[4]	A[5]	A[6]
1	0	1	0	0	0

寫出 B 和 CH 的最終值。

(2 分)

(b) 假設  $N = 6$ 。以下列 A 的初始內容執行 ALG1。

A[1]	A[2]	A[3]	A[4]	A[5]	A[6]
1	1	1	1	0	0

第 3 行會被執行多少次？

(1 分)

ALG2

行號	內容
1	$I \leftarrow 1$
2	$B \leftarrow 0$
3	當 $(I \leq N)$ AND $(B < N / 2)$ 執行
4	如果 $A[I] = 1$ 則
5	$B \leftarrow B + 1$
6	$I \leftarrow I + 1$
7	如果 $B \geq N / 2$
8	則 $CH \leftarrow 'X'$
9	否則 $CH \leftarrow 'Y'$

(c) 假設  $N = 6$ 。以下列 A 的初始內容執行 ALG2。

A[1]	A[2]	A[3]	A[4]	A[5]	A[6]
1	1	1	1	0	0

(i) 第 4 行會被執行多少次？

(1 分)

(ii) ALG2 比 ALG1 可更有效率。為甚麼？

(1 分)

**2022 #5 | 續**

(d) 完成下列 A 的初始內容，使語句「如果  $A[I] = 1$  則」在 ALG1 和 ALG2 兩者執行的次數相同。

A[1]	A[2]	A[3]	A[4]	A[5]	A[6]
1	1				

(2 分)

(e) ALG1 和以下 ALG3 應該總是產生相同的結果。完成以下第 3 行。

ALG3

行號 內容

1  $B \leftarrow 0$ 

2 設 I 由 1 至 N 執行

3  $B \leftarrow B + A[ \quad ]$ 4 如果  $B \geq N / 2$ 5 則  $CH \leftarrow 'X'$ 6 否則  $CH \leftarrow 'Y'$ 

(1 分)

(f) N 的值非常大。ALG1 和 ALG3 哪個算法較有效率？簡略說明。

(2 分)

**2023#2(a-c)**

伍老師使用以下試算表記錄學生在閱讀計劃的進展。截止日期和基本分數分別儲存在 C102 和 E102 內。

	A	B	C	D	E	F
1	StudID	書本編號	完成日期	等級	Valid	Score
2	S01	C001	31/12/2021	2	1	4
3	S29	C234	4/3/2022	3	0	2
4	S29	E456	1/1/2022	4	0	2
5	S03	C900	30/4/2021	5	1	7
6	S01	E233	1/12/2021	4	1	6
...	...	...	...	...	...	...
101						
102		截止日期	31/12/2021	基本分數	2	

(a) 如果學生能在 C102 的截止日期或之前完成閱讀書本，欄 E 會顯示數值 1，否則會顯示 0。在 E2 輸入公式，然後複製到 E3:E100。寫出 E2 的公式。

(2 分)

欄 F 內的數值是學生的分數，它們是根據 E102、欄 D 和欄 E 計算出來的，如下所示：

$$\text{Score} = \text{等級} \times \text{Valid} + \text{基本分數}$$

(b) 在 F2 輸入公式，然後複製到 F3:F100。寫出 F2 的公式。

(1 分)

(c) 伍老師用樞紐分析表去找出每位學生的分數總和。完成下列部份以建構樞紐分析表。

欄	值

(3 分)

### 2023#3ai, c, d

小芬編寫子程式，用以下偽代碼把陣列 A 排序。

行號	內容
1	$i \leftarrow 1$
2	$j \leftarrow 2$
3	當 $i < 5$ 執行
4	如果 $A[i] \geq A[i-1]$ 則
5	$i \leftarrow j$
6	$j \leftarrow j + 1$
7	否則
8	互換 $A[i]$ 和 $A[i-1]$ 的內容
9	$i \leftarrow i - 1$
10	如果 $i = 0$ 則
11	$i \leftarrow 1$

(a) 假設 A 的初始內容是：

A[0]	A[1]	A[2]	A[3]	A[4]
4	3	1	8	6

(i) TEMP 是一個變量。完成以下偽代碼去代表第 8 行。

TEMP  $\leftarrow$  A[i]

---



---

(1 分)



(c) 在第 10 行，若  $l_1 = 0$ ，改為  $l_1 \leq 0$ ，子程式的結果不會改變。為什麼？

(1 分)

(d) 以下三組測試數據，作為 A 的初始內容。選擇兩組去測試子程式，並簡略說明。

測試 1: -3, -1, 4, 0, 7  
測試 2: 1, 2, 3, 4, 5  
測試 3: 1, 1, 3, 3, 5

(2 分)

2023#4d

國華在購物中心經營一間玩具店。

(d) 國華利用機械人從貨倉提取玩具。機械人由 5 位元指令控制，以下顯示部份指令：

指令	描述
00000	停止
01111	開始
01001	順時針轉 90°
10000	逆時針轉 90°
11001	向前行一步
10111	向後行一步

(i) 數據傳輸時有機會出現一個錯誤位元。簡略描述如何查找出此錯誤。

(1 分)

(ii) 國華考慮使用 2 位元指令來控制機械人，如下所示：

指令	描述
00	停止
11	開始
10	順時針轉 90°
01	向前行一步

於下方寫出五個 2 位元指令，以執行「向後行一步」的行動。

11 \_\_\_\_\_ 00

(2 分)

(iii) 使用 2 位元指令而不是 5 位元指令來控制此種機械人有什麼好處？

(1 分)

2024#5

陣列 A 儲存了 6 個整數。算法 ALG1 和 ALG2 分別驗證下列要求 (I) 和 (II)。

- (I) A 中的所有整數是正數。
- (II) A 中的整數是以升序排列。

(a) 根據以上的要求，描述使用以下無效測試數據的目的：

情況 1:

A[0]	A[1]	A[2]	A[3]	A[4]	A[5]
3	3	5	5	8	5

目的：

情況 2:

A[0]	A[1]	A[2]	A[3]	A[4]	A[5]
-1	-1	3	5	5	8

目的：

(2 分)

(b) 如果 A 中的所有整數是正數，ALG1 便會輸出 TRUE，否則輸出 FALSE。完成以下 ALG1 的偽代碼。

P ← TRUE

設 i 由  至  執行

如果  則

P ← FALSE

輸出 P

(2 分)

(c) 如果 A 中的整數是以升序排列，ALG2 便會輸出 TRUE，否則輸出 FALSE。完成以下 ALG2 的偽代碼。

S ← TRUE

設 i 由  至  執行

如果  則

S ← FALSE

輸出 S

(2 分)

(d) 下列是算法 ALG3。

行號	內容
1	$C \leftarrow 1$
2	$M \leftarrow 1$
3	$N \leftarrow A[0]$
4	設 $i$ 由 1 至 5 執行
5	如果 $A[i-1] = A[i]$ 則
6	$C \leftarrow C + 1$
7	否則
8	$C \leftarrow 1$
9	如果 $C > M$ 則
10	$M \leftarrow C$
11	$N \leftarrow A[i]$
12	輸出 $N$

(i) 假設  $A$  的初始內容為：

$A[0]$	$A[1]$	$A[2]$	$A[3]$	$A[4]$	$A[5]$
3	5	5	5	8	8

完成循環（第 4 至 11 行）的第一、第二和最後一次迭代後， $C$ 、 $M$  和  $N$  的內容是甚麼？

	$C$	$M$	$N$
第一次迭代			
第二次迭代			
最後一次迭代			

(4 分)

(ii) ALG3 的目的是甚麼？