

香港電腦教育學會

THE HONG KONG ASSOCIATION FOR COMPUTER EDUCATION

資訊及通訊科技科模擬考試 2018

INFORMATION AND COMMUNICATION TECHNOLOGY
MOCK EXAMINATION 2018

試卷二 (D)

PAPER 2 (D)

評卷參考

MARKING SCHEME

中文版

CHINESE VERSION

(2018 01 09 版本)

©香港電腦教育學會 保留版權

The Hong Kong Association for Computer Education

All Rights Reserved 2018

1. (a) (i)

執行第二次步驟 1 的循環後，當 $c = 2$

A[0]	A[1]	A[2]	A[3]	A[4]	A[5]	A[6]	A[7]
5	6	27	24	36	12	15	20

1

執行第二次步驟 1 的循環後，當 $c = 3$

A[0]	A[1]	A[2]	A[3]	A[4]	A[5]	A[6]	A[7]
5	6	24	27	36	12	15	20

1

A 最後的內容

A[0]	A[1]	A[2]	A[3]	A[4]	A[5]	A[6]	A[7]
5	6	12	15	20	24	27	36

1

(ii) (1) 0

1

(2) 3

1

(iii) 將陣列 A 以小至大 排序

1

(iv) 不是。雖然結果相同，ALG2 需要更多的迭代次數。在 ALG2 中，內循環（步驟 2）需要迭代 c 次，但在 ALG1 中，內循環（步驟 3）就需要最多迭代 c 次。

1+1

(b) (i)

```

found ← 假
i ← 0
當 i ≤ n-1 與 非 found
    若 A[i] = UserInput 則
        found ← 真
    i ← i + 1
返回 found
(設-循環 / 將若-則條件放於“當”循環)
初始化 found          1
循環能正確完結        1
返回正確的值           1

```

其他方法 1:

```

found ← 假
設 i 由 0 至 n-1 執行
    若 A[i] = UserInput 則
        found ← 真
返回 found

```

其他方法 2:

```

i ← 0
當 i ≤ n-1 與 A[i] ≠ UserInput
    i ← i + 1
若 i = n 則 { A[i] ≠ UserInput 會導致超出範圍錯誤 }
    返回 假
否則
    返回 真 { 與整個若-則-否則改為 “返回 i > n” 相同 }

```

```

(ii) first ≤ last                                1
    found ← 真                                    1
    last ← mid - 1                                1
    first ← mid + 1                                1

```

```

2. (a) Page ← DIV( Index, R * C)                  1
      Row ← DIV( MOD(Index, R * C) , C)           1
      Col ← MOD( MOD(Index, R * C) , C)           1

```

(b) (i)

0	1	2	3	4	5	...	12	13	14	...
Call	Cam	Mail	Face	WApp	WApp	...	WApp	Rice	Song	

Face 正確 1, WApp 正確 1

```

(ii) 設 I 由 FromIndex 至 ToIndex+1                2
      (註: 接受 至 downto / to)
      Apps[I] ← Apps[I-1]                            2

```

```

其他方法: 設 I 由 FromIndex-1 至 ToIndex            2
           Apps[I+1] ← Apps[I]                      2

```

```

(c) 冒泡排序法:: 此算法容易實現                    1
      (須排序的項目數量少)
      合併排序法: 此算法的執行時間較短              1
      (須進行的比較次數較少)

```

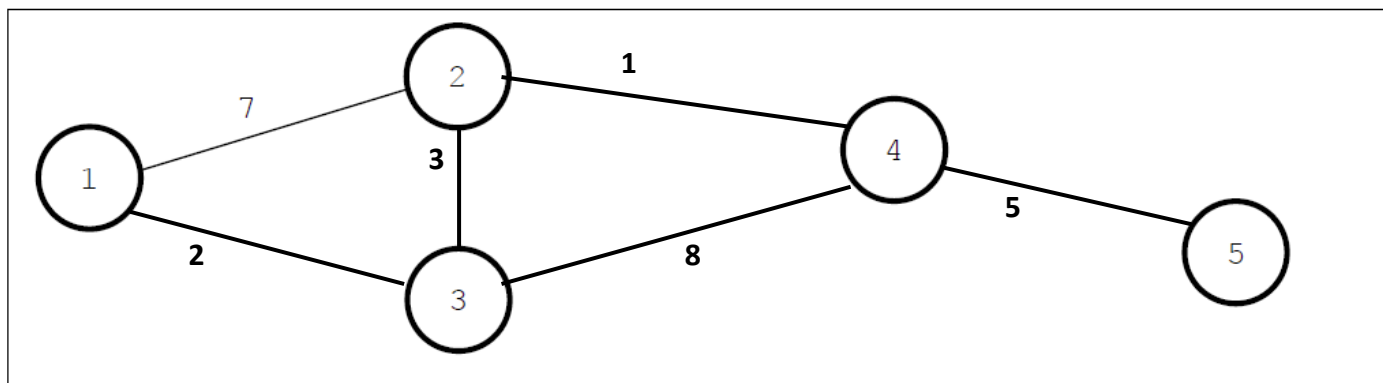
```

(d) (i) 瀑布模式: 較易管理, 因各階段是一個跟一個完成的 1
      前面的階段須達至完美才可進行下一階段
      (其他合理答案)
      應用系統迅速發展法(RAD): 開發時間相對較短    1
      (其他合理答案)
      (ii) 設計                                       1
           維修                                       1

```

3. (a) (i) $D_{13} = 2$, $D_{23} = 3$, $D_{24} = 1$, $D_{34} = 8$, $D_{45} = 5$

1 分: 任一對正確; 1 分: 全對



- (ii) 路徑 2: $1 \rightarrow 3 \rightarrow 4 \rightarrow 5$ 15 1
 路徑 3: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5$ 23 1
 路徑 4: $1 \rightarrow 3 \rightarrow 2 \rightarrow 4 \rightarrow 5$ 11 1
 從城市 1 去到城市 5 的最短路程長度: 11 1

(iii) 鏈表 1

- (b) (i) 組合語言是低階語言，可直接控制硬件，執行起來較快 (任一) 1
 邏輯語言是問題為本。 1
 物件導向語言促進代碼重用 1

- (ii) 第二代程式編寫語言: 匯編器，它將組合語言翻譯成機器語言 1
 第五代程式編寫語言: 解釋器，它將邏輯語句一句一句翻譯 1

或 編譯器

物件導向語言: 編譯器，它將程式整批翻譯成目標代碼 (任一) 1

或 解釋器

(c)

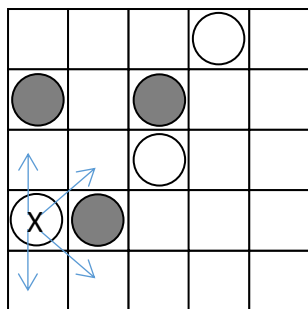
工作	所需時間	周									
		1	2	3	4	5	6	7	8	9	10
用戶要求規格	1 周	✓									
實施用戶界面	2 周		✓	✓							
實施主程式	3 周		✓	✓	✓						
單元測試	1 周					✓					
系統測試	1 周						✓				
用戶驗收測試	1 周							✓			

1

1

4. (a) 4 位置

1 分：任一正確；1 分：全對



(b) 當 $(0 \leq p + x \leq 4)$ 與 $(0 \leq q + y \leq 4)$

1

與 $(BD[p + x, q + y] = ' ')$ 執行以下 4 行

1

$BD[p + x, q + y] \leftarrow BD[p, q]$

1

$BD[p, q] \leftarrow ' '$

1

$p \leftarrow p + x$

1

$q \leftarrow q + y$

這棋子的移動應為 $B[p, q] \rightarrow B[p + y, q + x]$ 。

所以代碼應相應更改如下：(考生答案如上所列仍可得分。)

當 $(0 \leq p + y \leq 4)$ 與 $(0 \leq q + x \leq 4)$

1

與 $(BD[p + y, q + x] = ' ')$ 執行以下 4 行

1

$BD[p + y, q + x] \leftarrow BD[p, q]$

1

$BD[p, q] \leftarrow ' '$

1

$p \leftarrow p + y$

1

$q \leftarrow q + x$

c) q

1

$r \leftarrow r - 1$

1

3

1

d) (i) 行列對換以進行測檢

1

(ii) Reflect

設 i 由 0 至 4 執行

1 分：正確循環(外)

設 j 由 i + 1 至 4 執行

1 分：正確循環(內)

temp := B[i, j]

1 分：正確交換

B[i, j] := B[j, i]

1 分：全對

B[j, i] := temp