



福建中學

FUKIEN SECONDARY SCHOOL

中六畢業試 (2021-2022)

資訊及通訊科技

卷二 (D)

軟件開發

(一小時三十分)

日期：二零二二年一月二十七日

姓名：_____

時間：上午十一時至下午十二時三十分

班別：_____ 班號：_____

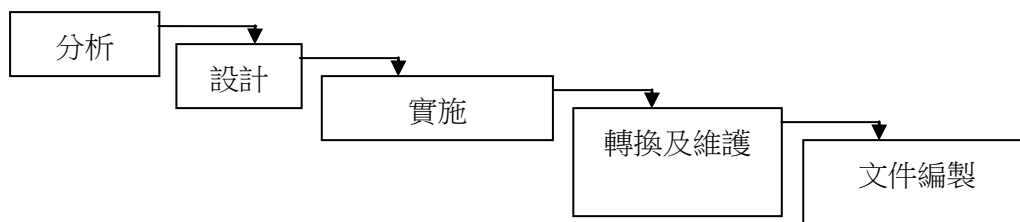
考生須知

1. 在適當位置填寫姓名、班別和班號。
2. 本試卷共設四題，**考生只須選答三題**。

只須選答三題。

1. 小芬是一位系統分析員。她為一家超級市場開發新的自助付款系統。

(a) 她打算利用瀑布模式來開發此系統。



(i) 指出**兩個不會**涉及用戶參與的階段。

(2 分)

(ii) 小芬打算在實施階段進行下列三種測試：

測試案例	詳細內容
1	超級市場管理員根據要求來檢查整個系統的功能。
2	測試掃描條碼能否取得產品 ID。
3	測試付款系統及庫存系統的整合。

以上三種測試類分別是什麼？

測試案例 1：_____

測試案例 2：_____

測試案例 3：_____

(3 分)

(iii) 高級編程語言和匯編語言均可用於系統開發。試為兩種語言分別舉出一個好處。

(2 分)

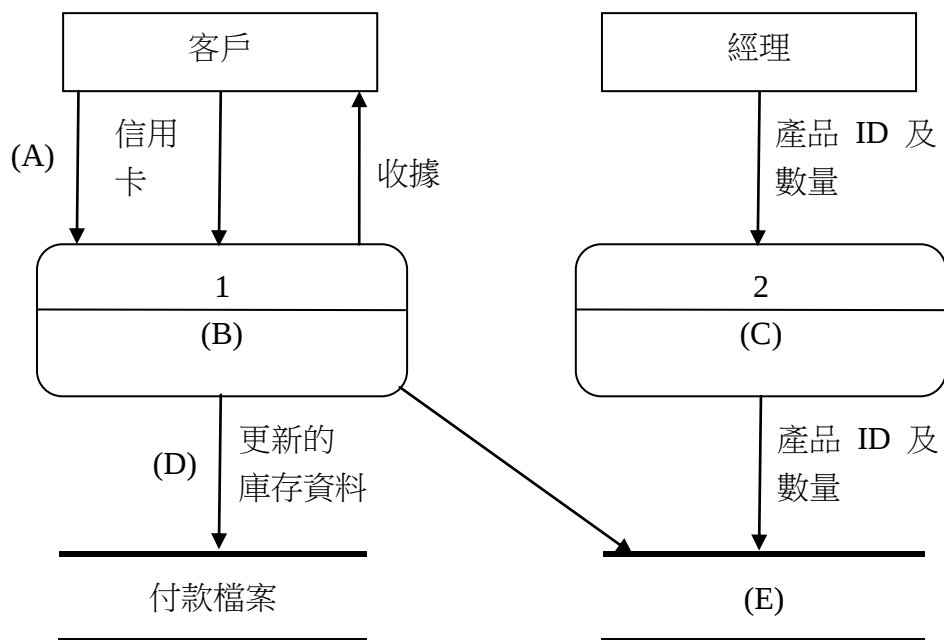
(iv) 湯母建議使用應用系統迅速發展法 (RAD) 取代瀑布模式來開發該系統。試舉出一個原因來支持他的說法。

(1 分)

- (b) 在設計階段，小芬運用了數據流程圖 (DFD) 來規劃超級市場整個系統的設計。

客戶可以使用自助付款系統來購買貨品。首先，客戶需掃描條碼以輸入產品 ID，然後出示用於付款的信用卡。自助付款系統會列印收據給客戶。同時，系統會利用付款訊息來更新付款檔案，並使用庫存信息來更新庫存檔案。

經理可以在庫存系統輸入產品 ID 及數量來更新庫存檔案。



- (i) 完成數據流程圖，並在下列空白位置寫上答案。

(A) _____

(B) _____

(C) _____

(D) _____

(E) _____

(5 分)

(ii) 除了數據流程圖，試舉出另一款可用於設計階段的圖表。

(1 分)

(iii) 小芬認為可以在用戶手冊內加入此圖表。你同意嗎？試解釋你的答案。

(1 分)

2. 小麗運用了鏈表去表示火車中多個車卡的連接，每一車卡由一英文字母命名，她用了平行陣列來實現鏈表。陣列 UNIT 記錄車卡名稱，陣列 NEXT 記錄下一個節點的位置，第一個節點儲存了 START。

(a) LL1

地址	UNIT	NEXT
0	START	4
1	A	-2
2	B	1
3	C	7
4	D	3
5	E	-2
6	F	2
7	G	6
8	H	5

(i) 除鏈表外，試舉出另一項數據結構及分析該數據結構在新增一個節點時的不同之處。

(ii) 在記錄下一個節點的位置中，-2 的用途是什麼？

(iii) 除了 -2，小麗還可以使用哪些其他的數字？

(iv) 順序列出在 START 之後的四個節點的內容。

(v) 除了第一個節點 START，鏈表中一共有多少個節點？

(7 分)

- (b) 假設鏈表中的節點順序為 $A \rightarrow F \rightarrow G \rightarrow C \rightarrow E \rightarrow B \rightarrow D \rightarrow H$ 。填寫以下 LL2 中的 NEXT 欄。

LL2

地址	UNIT	NEXT
0	START	1
1	A	
2	E	
3	F	
4	B	
5	D	
6	C	
7	G	
8	H	

(2 分)

- (c) 小麗設計了另一個結構，增加了另一個指示標 PREV，在每一個節點中，PREV 指向上一個節點。以下是新結構的一個例子：

LL3

地址	UNIT	PREV	NEXT
0	START	-1	4
1	A	2	5
2	E	6	1
3	F	4	7
4	B	0	3
5	D	1	8
6	C	7	2
7	G	3	6
8	H	5	-2

- (i) 試舉出一個在結構中加入 PREV 的好處。

(1 分)

- (ii) 小麗設計了一個指令 DELETE(Z) 用於移除位置 Z 的節點。當 DELETE(6) 執行完畢後，更新 LL3。

LL3

地址	UNIT	PREV	NEXT
0	START		
1			
2			
3			
4			
5			
6			
7			
8			

(3 分)

(iii) LL3 由平行陣列 UNIT、PREV 及 NEXT 組成。例如 UNIT[8] 是 H，PREV[8] 是 5，NEXT[8] 是 -2。假設 Z 並非鏈表中的開始及結束節點，完成以下 DELETE[Z] 指令的算法。

```
DELETE (Z) :  
NEXT [ PREV [ Z ] ] ← _____  
PREV [ NEXT [ Z ] ] ← _____
```

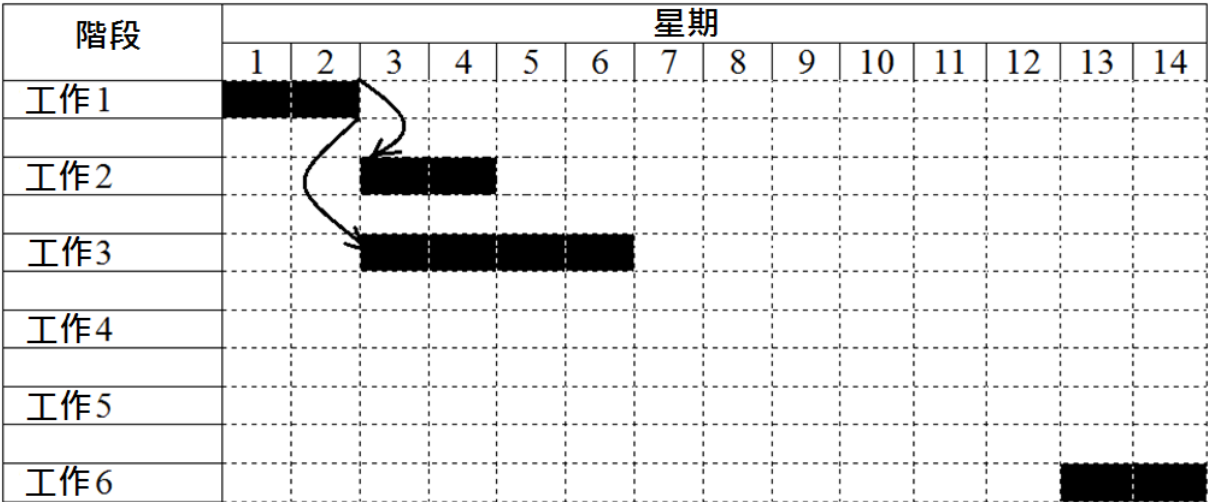
(2 分)

3. 瑪利是項目經理，她正在計劃建設成績計算系統。建設包括五項主要工作。以下圖表顯示每項工作所需時間及對其他工作的依賴。

工作	所需時間 (星期)	依賴其他工作
1	2	--
2	2	1
3	4	1
4	4	5
5	4	2
6	2	4,5

工作 1 是要求及分析，工作 2 及 3 是設計，工作 5 是實施，工作 6 是轉換。

(a) (i) 完成以下甘特圖：



(3 分)

(ii) 工作 4 處於系統開發周的甚麼階段？

(1 分)

(iii) 請列出項目的關鍵路徑。

(1 分)

(iv) 若工作 1、工作 2 及工作 4 減少一個星期完作工作，瑪利最少需要多少個星期才可完成這個項目？

(1 分)

(v) 瑪利決定採用瀑布模式而非快速應用程式開發 (RAD)。試列出一項使用瀑布模式的好處。

(1 分)

(b) 以下是算法 CAL1，整數陣列 A 的索引值由 0 至 $n-1$ 。

CAL1

步驟 1: $d \leftarrow 0$
 步驟 2: $e \leftarrow 0$
 步驟 3: 設 i 由 0 至 $n-1$ 執行步驟 4 至 10
 步驟 4: $c \leftarrow 0$
 步驟 5: 設 j 由 0 至 $n-1$ 執行步驟 6 至 7
 步驟 6: 如果 $A[i] = A[j]$ 則執行步驟 7
 步驟 7: $c \leftarrow c+1$
 步驟 8: 如果 $c > d$ 則執行步驟 9 至 10
 步驟 9: $d \leftarrow c$
 步驟 10: $e \leftarrow i$
 步驟 11: 輸出 $A[e]$

假設 $n = 10$ 。陣列 A 的初始值如下所示：

A[0]	A[1]	A[2]	A[3]	A[4]	A[5]	A[6]	A[7]	A[8]	A[9]
47	64	49	50	50	70	49	50	70	64

(i) 空運行 CAL1。試就以下情況，在完成步驟 4 至 10 後，填 d 及 e 的數值。

(1) $i = 1$

$d =$ _____ $e =$ _____

(1 分)

(2) $i = 4$

$d =$ _____ $e =$ _____

(1 分)

(ii) 當執行第三次步驟 3 的循環後，步驟 9 中的語句一共會被執行多少次？

_____ (1 分)

(iii) 算法 CAL1 的用途是甚麼？

_____ (1 分)

- (iv) 算法 CAL1 會被執行多次。你會選擇編譯語言還是直譯語言？試加以解釋。

(2 分)

- (v) 下列兩組測試用例將會用來測試 CAL1。

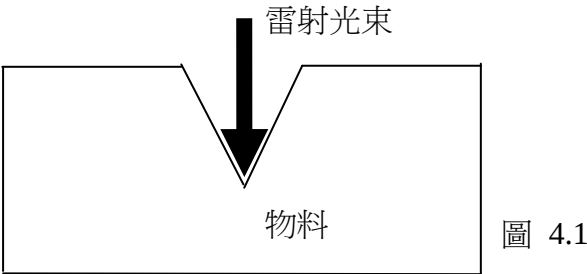
A 組	A[0]	A[1]	A[2]	A[3]	A[4]	A[5]	A[6]	A[7]	A[8]	A[9]
	47	47	47	47	47	47	47	47	47	47

B 組	A[0]	A[1]	A[2]	A[3]	A[4]	A[5]	A[6]	A[7]	A[8]	A[9]
	47	89	64	37	45	78	20	56	42	60

寫出及解釋這兩組測試用例的用途。

(2 分)

4. 雷射切割是一項利用雷射光束來分割物料的技术。雷射切割的原理是直接把高能量的光束射在物料上 (圖 4.1)。一個具有控制系統的驅動程式被用來控制雷射切割頭的移動和雷射光束的功率。不同功率的雷射光束用於不同的目的。低功率的雷射光束可以在材料上留下痕跡；高功率的雷射光束可以用來切割物料。



控制程式利用不同的子程式及變量來進行雷射切割的工序。

子程式	描述
LaserOn()	開啟雷射光束
LaserOff()	關閉雷射光束
MoveLH(x, y)	移動雷射切割頭至位置 (x, y)

變量	描述
i, j	用於循環的整數變量
x, y	一點的 x-坐標及 y-坐標

- (a) 子程式 DrawSquare1(x, y, length) 是用來繪畫一個填滿的正方形。該正方形左下角的點為 (x, y)，而其長度為 length。

完成 DrawSquare1(x, y, length) 的偽代碼。

```
DrawSquare1(x, y, length){
    MoveLH(x, y)

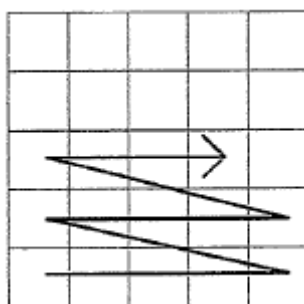
}
```

(3 分)

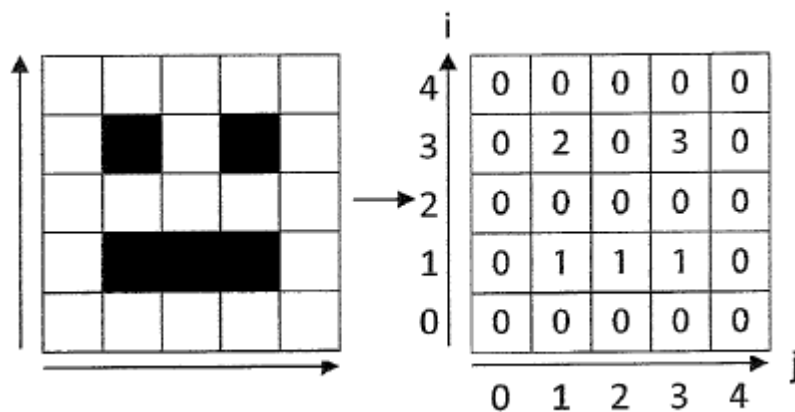
- (b) 在軟件開發的過程中，程式使用了子程式來提高模組化。指出兩個模組化程式設計的好處。

(2 分)

雷射切割頭以「之」字形的方式移動。



準備進行切割的圖案會貯存在一個圖像檔案中。驅動程式載入文件並將其貯存在整數陣列 `Pattern[i, j]` 之中。這些數字是切割順序號，表示要切割的順序。連續的切割點標記為相同的序列號。數字 0 用於表示不需要切割的點。



(圖 4.2)

- (c) 在切割圖案之前，驅動程式會載入圖像文件並將其轉換為陣列。圖像文件中的原始數據以線性形式表示。這對數據處理很有用，但不能直接地用於雷射切割。驅動程式讀取文件並將數據貯存在整數陣列

Image[] 中。就圖 4.2 的圖像文件，Image[] 所貯存的數據如下：

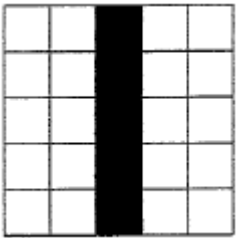
Image[i]	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	2	0	3	0	0	0	0	0	0	
i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24

(圖 4.3)

該圖案的尺寸為 5×5。在 Image[] 中，每 5 項會被映射到 Pattern[i, j] 的一列中。例如：Image[] 中的首 5 項（即 Image[0] 至 Image[4]）被映射到 Pattern[i, j] 的第一列，即 Pattern[0, 0]、Pattern[0,1]、Pattern[0, 2]、Pattern[0, 3] 及 Pattern[0, 4]，如圖 4.2 所示。

寫下 Image[] 在下列各圖案的內容。可以省略陣列元素為「0」的數組。

- (i) 圖像檔案：



Image[]:

Image[i]										
i	0	1	2	3	4	5	6	7	8	9

Image[i]										
i	10	11	12	13	14	15	16	17	18	19

Image[i]					
i	20	21	22	23	24

(ii) 圖像檔案：

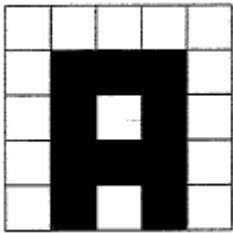


Image []:

Image[i]										
i	0	1	2	3	4	5	6	7	8	9

Image[i]										
i	10	11	12	13	14	15	16	17	18	19

Image[i]					
i	20	21	22	23	24

(2 分)

雷射切割器的控制系統具備更多變量：

變量	描述
Width, height	全程變量用來記錄所需切割的圖案的寬度和高度。
Pattern[i, j]	一個全程整數陣列以二維形式表達切割的圖案。
Image[i]	一個全程整數陣列以線性形式表達切割的圖案。
current	一個整數變量用來貯存當前的切割序號。
index	一個整數變量用來貯存 image[] 當下切割點的索引。

- (d) 子程式 mapping() 用來轉化二維陣列 Pattern[i, j] 為線性陣列 Image[]。

完成 mapping() 的偽代碼。

```
Mapping() {  
    設 j 由 0 至 _____ 執行  
        設 i 由 0 至 _____ 執行  
            _____ ← Pattern[i, j]
```

(2 分)

- (e) 子程式 `findNextPt(current, index)` 用作找出下一個在 `Image[]` 進行切割的點。`Image[index]` 是正在切割的點。整數變量 `current` 貯存著正在切割的順序號。

2	2	2	2	0
0	0	0	0	2
1	1	1	1	0
0	1	1	1	1
0	0	0	0	0

(圖 4.4)

圖 4.4 顯示，程式在切割序號 1 的最後一個方塊後，程式尋找以及移動到序號 2 的第一個方塊。

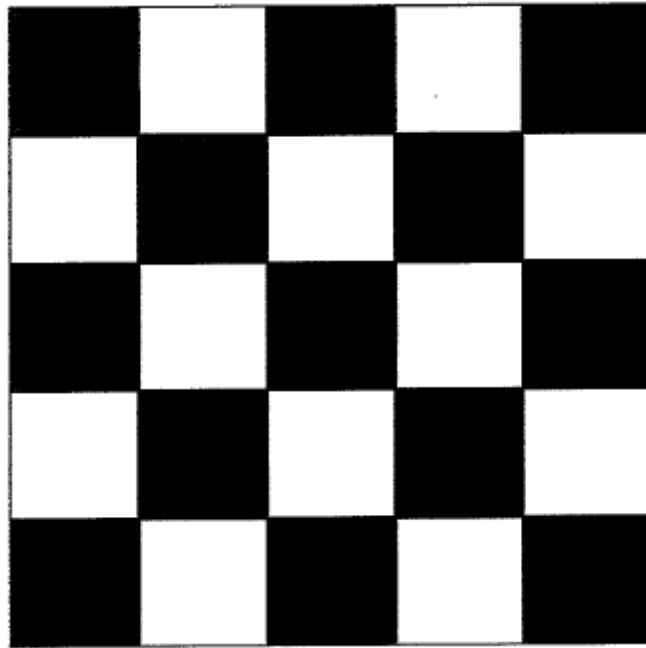
- (i) 完成 `findNextPt(current, index)` 的偽代碼。此程式利用線性檢索傳回下一個切割點的 `index`。如果未能找出下一個點，則程式會傳回 -1。

```
findNextPt(current, index){
```

```
}
```

(4 分)

(ii) 志明利用雷射切割器來切割以下圖案：



他指出於子程式 `findNextPt` 中，對分檢索比線性檢索更有效率。你認同他的說法嗎？解釋你的答案。

(2 分)

試卷完