

106年公務人員特種考試警察人員、一般警察人員考試及106年特種考試交通事業鐵路人員、退除役軍人轉任公務人員考試試題

代號：20140

全一張  
(正面)

考試別：一般警察人員考試

等別：二等考試

類科別：刑事警察人員數位鑑識組

科目：計算機系統(包括計算機結構、作業系統)

考試時間：2小時

座號：\_\_\_\_\_

※注意：(一)禁止使用電子計算器。

(二)不必抄題，作答時請將試題題號及答案依照順序寫在試卷上，於本試題上作答者，不予計分。

一、令  $F(a, b, c, d) = a'b'c'd' + a'b'cd' + a'bc'd + a'bcd + abc'd + abcd + ab'c'd' + ab'cd'$  為一具有四個輸入的布林函數。

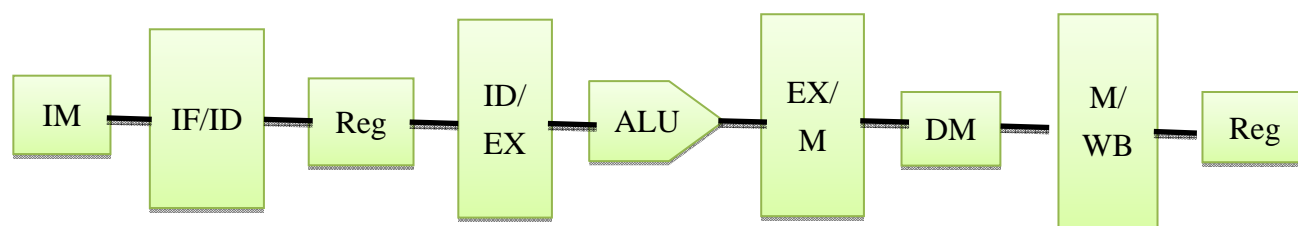
(一)應用卡諾圖 (Karnaugh Map)，化簡  $F(a, b, c, d)$ 。(5分)

(二)利用反及閘 (NAND) 來製作此化簡後的邏輯電路。(10分)

二、導管式計算機 (Pipeline Computer) 可以增加計算機執行指令的吞吐量 (Instruction Throughput)，但會形成三種障礙 (Hazard)，如結構障礙 (Structure Hazard)、控制障礙 (Control Hazard) 和資料障礙 (Data Hazard)。

(一)發生控制障礙時如何解決？(10分)

(二)一個典型的導管式計算機如圖一，由五個元件 (Component) 組成，如指令記憶體 (IM)、記錄器 (Reg) 讀取、運算單元 (ALU)、資料記憶體 (DM)、記錄器 (Reg) 寫入。每個元件在一個時序 (Clock Cycle) 完成，其中記錄器 (Reg) 讀取在時序的後半週完成而記錄器 (Reg) 寫入在時序的前半週完成。另一方面，元件之間有記錄器用來傳遞控制訊號和相關訊息，如指令讀取/指令解碼 (IF/ID)，指令解碼/指令執行 (ID/EX)，指令執行/資料存取 (EX/M) 和資料存取/記錄器寫入 (M/WB)。



圖一：五個元件的導管式計算機

當此計算機執行下面的程式時，說明它產生資料障礙的原因和解決方法。(15分)

```
sub $7, $1, $3 // Register 7 = Register 1 - Register 3 //
and $13, $7, $5 // Register 13 = Register 7 and Register 5 //
or $14, $6, $7 // Register 14 = Register 6 or Register 7 //
add $15, $7, $7 // Register 15 = Register 7 + Register 7 //
sw $16, 168($7) // Put the content of Register 16 back to memory based on Register 7 //
```

(請接背面)

106年公務人員特種考試警察人員、一般警察人員考試及106年特種考試交通事業鐵路人員、退除役軍人轉任公務人員考試試題

代號：20140

全一張  
(背面)

考試別：一般警察人員考試

等別：二等考試

類科別：刑事警察人員數位鑑識組

科目：計算機系統(包括計算機結構、作業系統)

三、詳述多執行序程式(Multithreaded Programming)的好處。(10分)

四、為減少下載用不到的分頁至主記憶體(Physical Memory)和降低交換時間(Swap Time),在虛擬記憶體管理(Virtual Memory Management)中,利用要求分頁(Demand Paging)技術來達成此目的。

(一)詳述虛擬記憶體管理(Virtual Memory Management)中,當要求分頁(Demand Paging)時發生了頁面錯誤(Page Fault),作業系統如何處理?(15分)

(二)令  $p$  ( $0 \leq p \leq 1$ ) 為發生頁面錯誤的機率,  $m_a$  為記憶體存取時間,  $pft$  為頁面錯誤處理時間,  $pft = 40000m_a$ , 則要求分頁的性能其有效的記憶體存取時間( $e_{fa}$ )為何?(5分)

(三)當分頁的性能只能小於 10% 時 ( $e_{fa} = 1.1m_a$ ), 則  $p$  應小於多少?(5分)

五、由於資源(Resources)有限,多個程序(Processes)在執行中會因競爭資源而造成死結(Deadlock)。

(一)詳述發生死結的四個必要條件。(10分)

(二)銀行家的算法(Banker's Algorithm)可以避免死結發生,令  $Max[i][j]=k$ , 表示程序  $P_i$  要求(Request)至多  $k$  個  $R_j$  類型的資源;  $Allocation[i][j]=k$ , 表示程序  $P_i$  分配到  $k$  個  $R_j$  類型的資源;  $Available[j]=k$ , 表示  $R_j$  類型的資源有  $k$  個;  $Need[i][j]=k$ , 表示程序  $P_i$  需要  $k$  個  $R_j$  類型的資源才可以完成工作。假設目前有 5 個行程分別為  $P_0$ 、 $P_1$ 、 $P_2$ 、 $P_3$ 、 $P_4$ , 和 3 種不同類型的資源分別為 A、B、C, 其中 A 類型的有 12 個、B 類型的有 5 個、C 類型的有 7 個。假設時間  $T_0$  時, 系統資源分配如表一, 詳述一程序序列(A Sequence of Processes)是當前分配狀態的安全序列(Safety Sequence)。當時間  $T_1$  時, 程序  $P_1$  額外要求 1 個 A、2 個 C, 詳述在這個情況下系統是否同意分配?(15分)

表一：分配狀態表

	Allocation			Max			Available		
	A	B	C	A	B	C	A	B	C
$P_0$	1	1	0	8	5	6	4	3	2
$P_1$	2	0	0	3	2	4			
$P_2$	3	0	2	10	0	2			
$P_3$	2	1	1	2	2	2			
$P_4$	0	0	2	4	3	5			