

題型一：為什麼組程式 (Assembler) 的設計方式常採用二次處理 (Two-Pass) 而不採用一次處理 (One-Pass)？

題型二：若程式中僅有少數向前參考 (Forward Reference) 的指令時，則稍微加以修改會比整個程式作第二回處理減輕很多負擔，為何有些組程式不使用單回處理以增加處理效率？

題型三：試問 IBM 360 組合語言有什麼特性？為何一定要建立一個兩回處理組程式 (2-Pass Assembler)？若建立三回處理程式 (3-Pass Assembler) 能再提供什麼額外的功能？

題型四：給予下列之組程式，並請回答其下的三個問題

```
SIMPLE  START
        BALR    15,0
        USING   *,15
LOOP    L      R1,TWO
        A      R1,TWO
        →
        ST     R1,FOUR
        CLI    FOUR+3,4
        BNE   LOOP
        BR    14
R1      EQU   1
TWO     DC   F'2'
FOUR    DS   F
        END
```

- (1) 寫出第一次處理結束後的符號表與文字表。
- (2) 寫出第二次處理時，基底暫存器表的改變情形。
- (3) 寫出第二次處理後的機器碼。

題型五：試問下列程式片段在組程式 (Assembler) 中需要多少回合 (Pass) 的處理。

```
A      EQU   B
B      EQU   C
C      EQU   999
```