組合語言 期中考　　110 年 11 月 18 日

寫下列題目中的組合語言程式可以使用的指令有：MOV, ADD, SUB, NEG, INC, DEC, JMP, JB, JNB, JA, JNA, JG, JNG, JL, JNL, JC, JO, JP, JS, JNC, JNS, JNO, JNS, JNP, JZ, JNZ, JCXZ, LOOP, LOOPZ, LOOPNZ, CALL, RET, PUSH, POP, AND, OR, NOT, XOR, TEST, CMP, STC, CLC 或課本第 6 章(包含)前，在課堂中講解過卻遺漏的其它指令。

可能用到的 ASCII 碼：ENTER 是 0DH，換行是 0AH，數字'0'是 30H，數字'1'是 31H，英文字母'A'是 41H，'a'是 61H，字母'B'是 42H，'b'是 62H，依此類推。

若覺得題目有錯誤，可說明錯誤處，並自行加以修改，而後解題。若題目無錯誤，而自行修改，當作是答錯。

1. 暫存器 AL 的存放的是一個代表一位數的十六進制數 ASCII 碼文字(即'0'~'F')。請寫出可將該文字轉變為對應的數值，並將結果放在暫存器 AL 的組合語言程式部分。文字中代表 10~15 的'A'~'F'可能是大寫，也可能是小寫。若文字正確，暫存器 AL 存放對應的數值。若不正確，則 AL 應存放 0FFH。例如，AL 原存有 31H('1')，則你的程式執行後，AL 應存有 1。若 AL 原存有 41H('A')，則結果的 AL 是 10。若 AL 原存有 50H('P') 則結果的 AL 是 0FFH。

2. 資料段(data segment)中定義了一個以字碼 0 作為結束的字串 TEXT1 和一個字碼 CH1，如下所示：

```
TEXT1 BYTE 200 dup (?)
CH1 BYTE ?
```

其中，由 TEXT1 起的記憶體位置已存放一些文字碼，並以數碼 0 當作結束。數碼 0 不算是字串的內容。位置 CH1 中則存放一個文字碼。請寫一程式片段，計算文字 CH1 在字串 TEXT1 中出現的次數，將結果放在暫存器 AL 中。

3. 資料的定義如下所示:：

```
TEXT1 BYTE 200 dup (?)
SIZE1 BYTE ?
POS1 BYTE ?
```

記憶體中名為 TEXT1 起的位置中，已經存有一個字串。字串的字數存放在位置 SIZE1 中。請寫一個程式，將這字串中的一個字移去，後面的字往前移。要移去的字位置存在 POS1 中。字移除後，新的字數也需存回 SIZE1。假設存在 POS1 內的值一定比存在 SIZE1 內的值小。

4. 資料的定義如下所示:：

```
MATX SDWORD 100 dup (?)
NOROW BYTE ?
NOCOL BYTE ?
```

```
ROWNO BYTE ?
COLNO BYTE ?
```

其中，MATX 起的位置存放著一個二維矩陣的內容，NOROW 和 NOCOL 依次存放這矩陣的列數和行數。矩陣內容分別依(0,0), (0,1), …, (1,0), (1,1),…, (2,0), …的順序，存放在記憶體中。請寫一程式片段，找出矩陣中第一個負數的位置。若找到，位置(列號和行號)應存到 ROWNO 和 COLNO 中。若找不到，ROWNO 和 COLNO 都應放 0FFH。假設 NOROW 和 NOCOL 內存的值都大於 1。

5. 請說明下列程式片段執行到最後一個指令的執行狀況，包括各指令的執行順序，和執行每一個指令後，暫存器(EAX、EBX、ECX 或 ESI)或記憶體位置的內容有甚麼改變。假設在執行第一個指令前，暫存器 EAX、EBX、ECX 和 ESI 的內容都是 0FFFFFFFFH。你的答案可以是個數值，如 CH 的內容改變為 12H；也可以是個未知值(地址或其它定義常數)，如 EBX 的內容改變為 VAR1+4。記憶體內容可寫成記憶體位置 VAR2 存 32 位元的 12345678H，或記憶體位置 VAR2+1 存 8 位元的 12H。

```
.DATA
TEXT1    BYTE   'Assembly Language Programming',0
VAR1     DWORD  010203H, 040506H, 070809H
.CODE
  XOR    ESI,ESI
  MOV    AL,TEXT1+10[ESI]
  MOVZX  ECX, BYTE PTR VAR1
  MOVZX  EBX, WORD PTR VAR1+4
  MOV    EDI, OFFSET VAR1
L1:
  OR     BYTE PTR [EDI+8], 30H
  INC    TEXT1[ESI]
  INC    ESI
  ADD    EDI, 1
  LOOP   L1
  ADD    AH, 1
```