课堂练习题

1. 以下程序段的功能是将（BX）中的内容以十六进制串的形式显示到屏幕上，请在程序空白处填入空缺的代码。

```
F2T16    PROC
             PUSH     AX
             PUSH     CX
             PUSH     DX
             _____________
F2T16_L1：ROL     BX，  4
             MOV      DX，  BX
             AND      DX，0FH
             CMP      DL，9
             ____________
             ADD      DL，   7
F2T16_L2：ADD     DL，30H
             _____________
             INT      21H
             DEC      CX
             __________
             POP      DX
             POP      CX
             POP      AX
             ________________
F2T16    ENDP
```

【解答】

```
F2T16    PROC
             PUSH     AX
             PUSH     CX
             PUSH     DX
             _____________ MOV CX, 4 ——①
F2T16_L1：ROL     BX，  4
             MOV      DX，  BX
             AND      DX，0FH
             CMP      DL，9
             ____________JB  F2T16_L2 ——②
             ADD      DL，   7
F2T16_L2：ADD     DL，30H
             _____________MOV AH, 2——③
             INT      21H
             DEC      CX
             __________JNE  F2T10_L1——④
             POP      DX
             POP      CX
             POP      AX
             ________________RET——⑤
```

```
F2T16     ENDP
```
2. 编写子程序，使其能够显示 CALL 指令下面的串

```
.386
STACK  SEGMENT USE16 STACK
                DB 200 DUP(0)
STACK  ENDS
CODE   SEGMENT USE16
                ASSUME  CS:CODE,SS:STACK
BEGIN:
                   CALL  DISPLAY
                   DB    'HELLO',0DH,0AH,0
                   CALL  DISPLAY
                    DB    'VERY GOOD',0DH, 0AH,0
                    MOV  AH,4CH
                    INT   21H
DISPLAY  PROC
            ……
DISPLAY  ENDP
CODE  ENDS
END  BEGIN
```

【解答】

此题首先要明确的是从主程序进入到子程序后，堆栈栈顶存放的是字符串'HELLO',0DH,0AH,0 的第一个字符'H'的地址。因为在执行 CALL 指令时，指令指示器 IP 总是指向 CALL 指令之后的存储单元的地址，并且在执行 CALL 指令时，将这个地址放到栈顶，进入子程序。因此在子程序中要做的工作就是从栈顶得到待显示的字符的地址，然后采用 2 号调用显示所有的字符串，直到遇到空白字符 0 为止。然后在执行 RET 指令之前，将字符串之后的指令的地址放到栈顶，使得执行 RET 之后正确返回到字符串后的指令执行。程序段如下：

```
DISPLAY   PROC
          POP       BX;
   LL1:   MOV       DL, CS:[BX]
          INC       BX
          CMP   DL, 0
          JE     EX1
          MOV       AH, 2
          INT    21H
          JMP       LL1
EX1:      PUSH  BX
          RET
DISPLAY  ENDP
```

本题也可以采用 9 号调用来完成，但是需要设置好 DS 段寄存器的内容与 CS 一致。因为 9 号调用中要求 DX 中保存的待显示字符串首地址必须在 DS 段中，其次要将 0 转换成'$'。

3.改错题。（共 15 个问题）

下列程序的功能是：比较存放在 BUF1 和 BUF2 中两个串长均为 N 的字符串是否相等，相等则显示“MATCH!”,不相等则显示“NO MATCH!”,并将碰到的第一对不相同的字符存放到 BUF3 中。请将下列程序中的语法错误和逻辑错误圈出来并在其右侧写出正确的形式。

```
.386
STACK  SEGMENT  USE16  STACK
         200 DUP(0)
STACK  ENDS
DATA  SEGMENT  USE16
    BUF1  DB  ‘MOV  AX,BX’
    BUF2  DB  ‘MOV  AX,CX’
    N     DB  ‘$’-BUF2
    BUF3  DB  0,0
    STR1  DB  0DH,0AH,‘MATCH!’
    STR2  DB  0D,0AH, ‘NO MATCH!$’
DATA  ENDS
CODE  SEGMENT
     ASSUME  CS：CODE，  DS：DATA, SS：STACK
     BEGIN：   MOV  DS, DATA
               LEA  SI,BUF1
     NEXT:  MOV  BP,0
               MOV  CX,N
               MOV  AL,[SI+BP]
               CMP  AL, BUF1+N[BP]
               JNE   NOMA
               LOOP   NEXT
               LEA  DX,STR1
      NOMA:  MOV  BUF3,AL
               MOV  BUF3+1, BUF2[BP]
               MOV  DX, STR2
      C9:   MOV  AH,9
               INT  21H
               MOV  AH, 4CH
               INT  21H
     CODE   ENDS
END
```

【解答】

```
.386
STACK  SEGMENT  USE16  STACK
         200 DUP(0)                DB
STACK  ENDS
DATA  SEGMENT  USE16
```

```
    BUF1  DB  'MOV  AX,BX'
    BUF2  DB  'MOV  AX,CX'
    N     DB  '$'-BUF2              $
    BUF3  DB  0,0
    STR1  DB  0DH,0AH,'MATCH!'        'MATCH! $'
    STR2  DB  0D,0AH, 'NO MATCH!$'      0DH
DATA  ENDS
CODE  SEGMENT                     USE16
      ASSUME  CS：CODE，  DS：DATA, SS：STACK
      BEGIN：  MOV  DS, DATA        MOV  AX,DATA      MOV  DS,AX
               LEA  SI,BUF1
      NEXT:  MOV  BP,0              去掉标号，移下两行
               MOV  CX,N              MOVZX   CX,N
               MOV  AL,[SI+BP]          NEXT: MOV AL,DS:[SI+BP]
               CMP  AL, BUF1+N[BP]        BUF2[BP]
               JNE   NOMA             下一行加上  INC  BP
               LOOP   NEXT
               LEA  DX,STR1             下一行加上  JMP  C9
      NOMA:  MOV  BUF3,AL
               MOV  BUF3+1,BUF2[BP]   MOV  AL,BUF2[BP]   MOV  BUF3+1,AL
               MOV  DX,STR2             LEA
      C9:   MOV  AH,9
               INT  21H
               MOV  AH,4CH
               INT  21H
     CODE   ENDS
              END                   BEGIN
```

要说明的是，若 N DB '$'-BUF2 的修改为：

N = $-BUF2，则该题只有 14 个错误。此时指令 MOV CX,N 和 CMP AL, BUF1+N[BP]均为正确的指令。