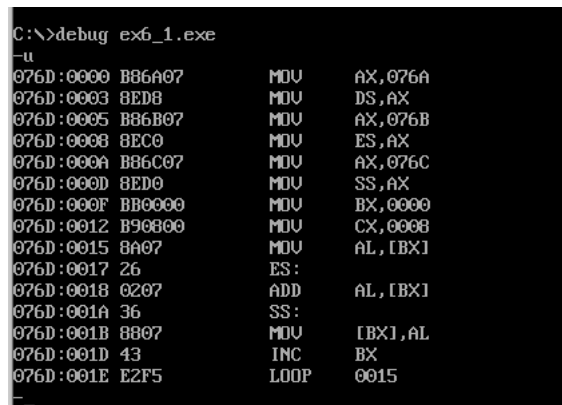


实验6 参考答案及评分标准

1. 程序如下，编写 code 段中代码，将 a 段和 b 段中的数据依次相加，将结果存到 C 段中。（50 分）

```
assume cs:code
a segment
    db 1,2,3,4,5,6,7,8
a ends
b segment
    db 1,2,3,4,5,6,7,8
b ends
c segment
    db 0,0,0,0,0,0,0,0
c ends
code segment
start:
    mov ax,a
    mov ds,ax
    mov ax,b
    mov es,ax
    mov ax,c
    mov ss,ax
    mov bx,0
    mov cx,8
s:    mov al,[bx]
        mov ss:[bx],al
        mov al,es:[bx]
        add ss:[bx],al
        inc bx
        loop s
    mov ax,4c00h
    int 21h
code ends
end start
```

核心代码测试的相关截图如下：



```
C:\>debug ex6_1.exe
-u
076D:0000 B86A07      MOV     AX,076A
076D:0003 8ED8          MOV     DS,AX
076D:0005 B86B07      MOV     AX,076B
076D:0008 8EC0          MOV     ES,AX
076D:000A B86C07      MOV     AX,076C
076D:000D 8ED0          MOV     SS,AX
076D:000F BB0000      MOV     BX,0000
076D:0012 B90800      MOV     CX,0008
076D:0015 8A07          MOV     AL,[BX]
076D:0017 26          ES:    INC     BX
076D:0018 0207      ADD     AL,[BX]
076D:001A 36          SS:    MOV     [BX],AL
076D:001B 8B07      MOV     [BX],AL
076D:001D 43          INC     BX
076D:001E E2F5      LOOP    0015
```

```

-u
076D:0020 B8004C      MOV     AX,4C00
076D:0023 CD21        INT     21
076D:0025 8946FC      MOV     [BP-04],AX
076D:0028 40           INC     AX
076D:0029 7520        JNZ     004B
076D:002B 833E130218  CMP     WORD PTR [0213],+18
076D:0030 7513        JNZ     0045
076D:0032 EB0B3C      CALL    3C40
076D:0035 B88001      MOV     AX,0180
076D:0038 50           PUSH    AX
076D:0039 FF7604      PUSH    [BP+04]
076D:003C EB3973      CALL    737B
076D:003F 83C404      ADD     SP,+04

```

```

-g 20
AX=0710 BX=0008 CX=0000 DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=076A ES=076B SS=076C CS=076D IP=0020  NU UP EI PL NZ NA PO NC
076D:0020 B8004C      MOV     AX,4C00
-a ds:0
076A:0000 01 02 03 04 05 06 07 08 00 00 00 00 00 00 00 00 .....
076A:0010 01 02 03 04 05 06 07 08 00 00 00 00 00 00 00 00 .....
076A:0020 02 04 06 08 0A 0C 0E 10 00 00 00 00 00 00 00 00 .....
076A:0030 B8 6A 07 8E D8 B8 6B 07 8E C0 B8 6C 07 8E D0 BB .j....k....l...
076A:0040 00 00 B9 08 00 8A 07 26 02 07 36 88 07 43 E2 F5 .....&..6..C..
076A:0050 B8 00 4C CD 21 89 46 FC 40 75 20 83 3E 13 02 18 ..L!.F.eu .>...
076A:0060 75 13 EB 0B 3C B8 80 01 50 FF 76 04 E8 39 73 83 u...<...P.v..9s.
076A:0070 C4 04 89 46 FC 83 7E FC FF 74 C5 FF 76 FC E8 1B ...F...t..v...

```

由上图可见，076a:0 开始的那一行对应的是 a 段；076a:10 开始的那一行是 b 段；076a:20h 开始的那一行是 c 段，并且结果是正确的，c 段中前 8 个元素中的每一个元素正好是 a 段和 b 段对应元素相加之和。

2. 程序如下，编写 code 段中代码，用 PUSH 指令将 A 段中的前 8 个字型数据，逆序存储到 B 段中。（50 分）

```
assume cs:code
```

```
a segment
```

```
dw 1,2,3,4,5,6,7,8,9, 0ah,0bh,0ch,0dh,0eh,0fh,0ffh
```

```
a ends
```

```
b segment
```

```
dw 0,0,0,0,0,0,0,0
```

```
b ends
```

```
code segment
```

```
start: mov ax,a
```

```
mov ds,ax ;ds 指向 a 段
```

```
mov ax,b
```

```
mov bx,0 ;ds:bx 指向 a 段的第 1 个单元
```

```
mov ss,ax
```

```
mov sp,16 ;设置栈顶指向 b:16
```

```
mov cx,8
```

```
s: push [bx]
```

```
add bx,2
```

```
loop s ;将 a 段中 0~16 个单元逆次入栈
```

```
mov ax,4c00h
```

```
int 21h
```

```
code ends
```

```
end start
```

核心代码测试的相关结果截图如下

```
C:\>debug ex6_2.exe
```

```
-u
```

```
076D:0000 B86A07      MOV     AX,076A
076D:0003 8ED8          MOV     DS,AX
076D:0005 B86C07      MOV     AX,076C
076D:0008 B80000      MOV     BX,0000
076D:000B 8ED0          MOV     SS,AX
076D:000D BC1000      MOV     SP,0010
076D:0010 B90800      MOV     CX,0008
076D:0013 FF37      PUSH    [BX]
076D:0015 83C302      ADD     BX,+02
076D:0018 E2F9      LOOP    0013
076D:001A B8004C      MOV     AX,4C00
076D:001D CD21      INT     21
076D:001F EB5673      CALL    7378
```

```
-g 1a
```

```
AX=076C BX=0010 CX=0000 DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=076A ES=075A SS=076C CS=076D IP=001A  NU UP EI PL NZ AC PO NC
076D:001A B8004C      MOV     AX,4C00
```

```
-d ds:0
```

```
076A:0000  01 00 02 00 03 00 04 00-05 00 06 00 07 00 08 00  .....
076A:0010  09 00 0A 00 0B 00 0C 00-0D 00 0E 00 0F 00 FF 00  .....
076A:0020  08 00 07 00 06 00 05 00-04 00 03 00 02 00 01 00  .....
076A:0030  B8 6A 07 BE D8 B8 6C 07-BB 00 00 8E D0 BC 10 00  .j...l.....
076A:0040  B9 08 00 FF 37 83 C3 02-E2 F9 B8 00 4C CD 21 E8  ....7.....L?!
076A:0050  56 73 83 C4 04 89 46 FC-40 75 20 83 3E 13 02 18  Us....F.eu .>...
076A:0060  75 13 EB 0B 3C B8 80 01-50 FF 76 04 E8 39 73 83  u...<...P.v..9s.
076A:0070  C4 04 89 46 FC 83 7E FC-FF 74 C5 FF 76 FC E8 1B  ...F...t.v...
```

从上述截图可以发现，076a:0 开始的前两行是 a 段，076a:20 开始的那一行是 b 段，并且 b 段的内容的前 8 个字正好是 a 段内容的前 8 个字逆序存储结果。