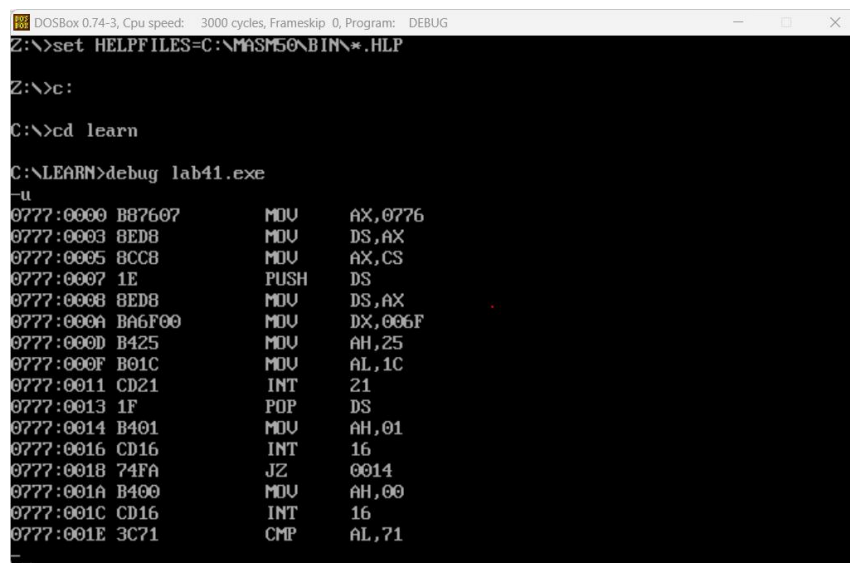


第 4 次上机

班级	学号	姓名
计试 2201	2223211946	王顺平

1、中断程序设计

(1) 反汇编的截图



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip: 0, Program: DEBUG
Z:\>set HELPFFILES=C:\MASM60\BIN\*.HLP
Z:\>c:
C:\>cd learn
C:\LEARN>debug lab41.exe
-u
0777:0000 B87607      MOV     AX,0776
0777:0003 8ED8      MOV     DS,AX
0777:0005 8CC8      MOV     AX,CS
0777:0007 1E        PUSH    DS
0777:0008 8ED8      MOV     DS,AX
0777:000A BA6F00      MOV     DX,006F
0777:000D B425      MOV     AH,25
0777:000F B01C      MOV     AL,1C
0777:0011 CD21      INT     21
0777:0013 1F        POP     DS
0777:0014 B401      MOV     AH,01
0777:0016 CD16      INT     16
0777:0018 74FA      JZ      0014
0777:001A B400      MOV     AH,00
0777:001C CD16      INT     16
0777:001E 3C71      CMP     AL,71
```

(2) 在进行计算前，显示 ID、NUM 的内存值的截图（多显示、少显示均扣分）

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip: 0, Program: DEBUG
0777:0011 CD21      INT     21
0777:0013 1F        POP     DS
0777:0014 B401      MOV     AH,01
0777:0016 CD16      INT     16
0777:0018 74FA      JZ      0014
0777:001A B400      MOV     AH,00
0777:001C CD16      INT     16
0777:001E 3C71      CMP     AL,71
-t
AX=0776 BX=0000 CX=01F8 DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0766 ES=0766 SS=0775 CS=0777 IP=0003  NU UP EI PL NZ NA PO NC
0777:0003 8ED8      MOV     DS,AX
-t
AX=0776 BX=0000 CX=01F8 DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0776 ES=0766 SS=0775 CS=0777 IP=0005  NU UP EI PL NZ NA PO NC
0777:0005 BCCB      MOV     AX,CS
-d 0 e
0776:0000 32 32 32 33 32 31 31 39-34 36 00 00 00 00 00 2223211946.....
-d 0 d
0776:0000 32 32 32 33 32 31 31 39-34 36 00 00 00 00 00 2223211946....
-d 0 b
0776:0000 32 32 32 33 32 31 31 39-34 36 00 00 2223211946..

```

(3) 运行到返回 dos 前暂停，对屏幕显示的输出结果（NUM 值的对应的 ASCII 字符串）截图【结果要与步骤（4）中的内存值一致】

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip: 0, Program: DEBUG
0777:0068 B402      MOV     AH,02
0777:006A CD21      INT     21
0777:006C B80000     MOV     AX,0000
0777:006F A00A00     MOV     AL,[000A]
-u 6a
0777:006A CD21      INT     21
0777:006C B80000     MOV     AX,0000
0777:006F A00A00     MOV     AL,[000A]
0777:0072 240F      AND     AL,0F
0777:0074 3C0A      CMP     AL,0A
0777:0076 7C02      JL      007A
0777:0078 0427      ADD     AL,27
0777:007A 0430      ADD     AL,30
0777:007C 8AD0      MOV     DL,AL
0777:007E B402      MOV     AH,02
0777:0080 CD21      INT     21
0777:0082 B8004C     MOV     AX,4C00
0777:0085 CD21      INT     21
0777:0087 FF060A00    INC     WORD PTR [000A]
-g 0082
0011
AX=0231 BX=0000 CX=0204 DX=0031 SP=0000 BP=0000 SI=0000 DI=0000
DS=0776 ES=0766 SS=0775 CS=0777 IP=0082  NU UP EI PL NZ NA PO NC
0777:0082 B8004C     MOV     AX,4C00

```

(4) 在完成步骤（3）操作后，立即显示 ID、NUM 的内存值的截图（多显示、少显示均扣 分）

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip: 0, Program: DEBUG
-u 6a
0777:006A CD21      INT     21
0777:006C B80000     MOV     AX,0000
0777:006F A00A00     MOV     AL,[000A]
0777:0072 240F      AND     AL,0F
0777:0074 3C0A      CMP     AL,0A
0777:0076 7C02      JL      007A
0777:0078 0427      ADD     AL,27
0777:007A 0430      ADD     AL,30
0777:007C 8AD0      MOV     DL,AL
0777:007E B402      MOV     AH,02
0777:0080 CD21      INT     21
0777:0082 B8004C     MOV     AX,4C00
0777:0085 CD21      INT     21
0777:0087 FF060A00    INC     WORD PTR [000A]
-g 008Z
0011
AX=0231 BX=0000 CX=0204 DX=0031 SP=0000 BP=0000 SI=0000 DI=0000
DS=0776 ES=0766 SS=0775 CS=0777 IP=0082  NV UP EI PL NZ NA PO NC
0777:0082 B8004C     MOV     AX,4C00
-d 0 d
0776:0000 32 32 32 33 32 31 31 39-34 36 11 00 00 00      2223211946....
-d 0 b
0776:0000 32 32 32 33 32 31 31 39-34 36 11 00      2223211946..

```

(5) 源代码

```

1.  data segment
2.      id db '2223211946'
3.      num dd 0000h
4.  data ends
5.
6.  code segment
7.      assume ds:data,cs:code
8.
9.  start: mov ax,data
10.         mov ds,ax
11.
12.         mov ax,cs
13.         push ds
14.         mov ds,ax
15.         mov dx,offset int1c
16.
17.         mov ah,25h
18.         mov al,1ch
19.         int 21h
20.         pop ds
21.
22.
23.     waitn: mov ah,1
24.         int 16h
25.         jz waitn

```

```
26.      mov ah,0
27.      int 16h
28.      cmp al,'q'
29.      jne waitn
30.
31.      mov cl,4
32.      mov ax,0
33.      mov al,byte ptr [num + 1]
34.      shr ax,cl
35.      and al,0Fh
36.      cmp al,0ah
37.      jl j1
38.      add al,27h
39.      j1: add al,30h
40.      mov dl,al
41.      mov ah,2
42.      int 21h
43.
44.      mov ax,0
45.      mov al,byte ptr [num + 1]
46.      and al,0Fh
47.      cmp al,0ah
48.      jl j2
49.      add al,27h
50.      j2: add al,30h
51.      mov dl,al
52.      mov ah,2
53.      int 21h
54.
55.      mov cl,4
56.      mov ax,0
57.      mov al,byte ptr [num]
58.      shr ax,cl
59.      and al,0Fh
60.      cmp al,0ah
61.      jl j3
62.      add al,27h
63.      j3: add al,30h
64.      mov dl,al
65.      mov ah,2
66.      int 21h
67.
68.      mov ax,0
69.      mov al,byte ptr [num]
```

```
70.         and al,0Fh
71.         cmp al,0ah
72.         jl j4
73.         add al,27h
74.   j4: add al,30h
75.         mov dl,al
76.         mov ah,2
77.         int 21h
78.
79.
80.         mov ax,4c00h
81.         int 21h
82.
83. int1c  proc   near
84.         inc word ptr [num]
85.         iret
86. int1c  endp
87.
88. code ends
89. end start
```

3、BIOS 和 DOS 中断

(1) 反汇编的截图

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
C:\LEARN>debug lab42.exe.
Extended Error 2

-q
^ Error
-q

C:\LEARN>debug lab42.exe
-u
0780:0000 B87E07      MOV     AX,077E
0780:0003 8ED8        MOV     DS,AX
0780:0005 BF0A00      MOV     DI,000A
0780:0008 B400        MOV     AH,00
0780:000A CD16      INT     16
0780:000C 3C0D      CMP     AL,0D
0780:000E 740B      JZ      001B
0780:0010 BB1400      MOV     BX,0014
0780:0013 2C30      SUB     AL,30
0780:0015 D7        XLAT
0780:0016 8B05      MOV     [DI],AL
0780:0018 47        INC     DI
0780:0019 EBED      JMP     000B
0780:001B B8004C      MOV     AX,4C00
0780:001E CD21      INT     21
```

(2) 在进行计算前，显示 ID、BUFFER 的内存值的截图（多显示、少显示均扣分）

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
0780:0010 BB1400      MOV     BX,0014
0780:0013 2C30      SUB     AL,30
0780:0015 D7        XLAT
0780:0016 8B05      MOV     [DI],AL
0780:0018 47        INC     DI
0780:0019 EBED      JMP     000B
0780:001B B8004C      MOV     AX,4C00
0780:001E CD21      INT     21
-t
AX=077E BX=0000 CX=0129 DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=076E ES=076E SS=077D CS=0780 IP=0003  NU UP EI PL NZ NA PO NC
0780:0003 8ED8      MOV     DS,AX
-d 0 13
076E:0000 CD 20 FF 9F 00 EA FF FF-AD DE 4F 03 AD 01 8A 03  . . . . .0 . . . .
076E:0010 AD 01 17 03  . . . .
-t
AX=077E BX=0000 CX=0129 DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=077E ES=076E SS=077D CS=0780 IP=0005  NU UP EI PL NZ NA PO NC
0780:0005 BF0A00      MOV     DI,000A
-d 0 13
077E:0000 32 32 32 33 32 31 31 39-34 36 00 00 00 00 00 00  2223211946 . . . .
077E:0010 00 00 00 00  . . . .
```

(3) 输入回车后，显示 ID、BUFFER 的内存值的截图（多显示、少显示均扣分）

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip: 0, Program: DEBUG
0780:0008 B400      MOV     AH,00
0780:000A CD16      INT     16
0780:000C 3C0D      CMP     AL,0D
0780:000E 740B      JZ      001B
0780:0010 BB1400     MOV     BX,0014
0780:0013 2C30      SUB     AL,30
0780:0015 D7        XLAT
0780:0016 8B05      MOV     [DI],AL
0780:0018 47        INC     DI
0780:0019 EBED      JMP     000B
0780:001B B8004C     MOV     AX,4C00
0780:001E CD21      INT     21
0780:0020 4E        DEC     SI
0780:0021 42        INC     DX
0780:0022 3030      XOR     [BX+SI],DH
0780:0024 B700      MOV     BH,00
-g 001b
AX=1C0D BX=0014 CX=0129 DX=0000 SP=0000 BP=0000 SI=0000 DI=0014
DS=077E ES=076E SS=077D CS=0780 IP=001B NU UP EI PL ZR NA PE NC
0780:001B B8004C      MOV     AX,4C00
-d 0 13
077E:0000 32 32 32 33 32 31 31 39-34 36 09 09 09 01 09 05 2223211946.....
077E:0010 05 04 03 08 .....

```

(4) 源代码

```

1.      data segment
2.      id db '2223211946'
3.      buffer db 10 dup (?)
4.      lbel db 7,5,9,1,3,6,8,0,2,4
5.      data ends
6.
7.      code segment
8.      assume ds:data,cs:code
9.
10.     start: mov ax,data
11.           mov ds,ax
12.
13.           mov di,offset buffer
14.     lop:  mov ah,0
15.           int 16h
16.           cmp al,0dh
17.           je quit
18.           mov bx,offset lbel
19.           sub al,30h
20.           xlat
21.           mov [di],al
22.           inc di
23.           jmp lop
24.
25.

```

```
26. quit:  mov ax,4c00h
```

```
27.         int 21h
```

```
28. code ends
```

```
29. end start
```