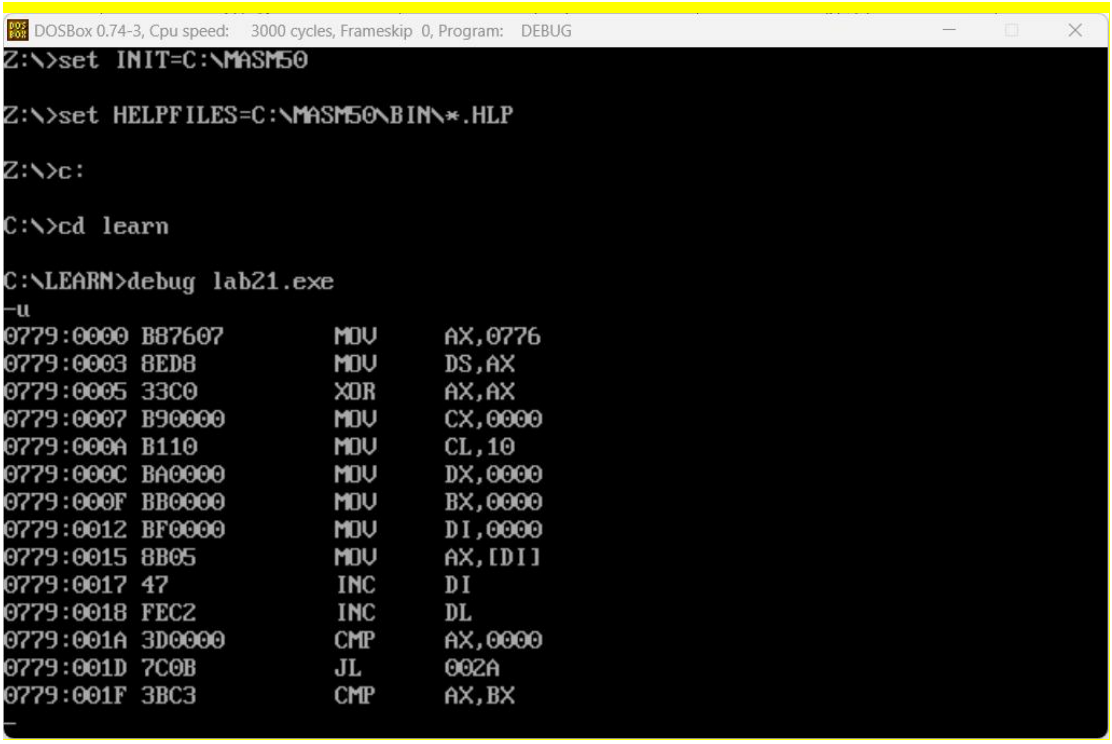
第 2 次上机

班级	学号	姓名
计试 2201	2223211946	王顺平

1、 循环程序设计

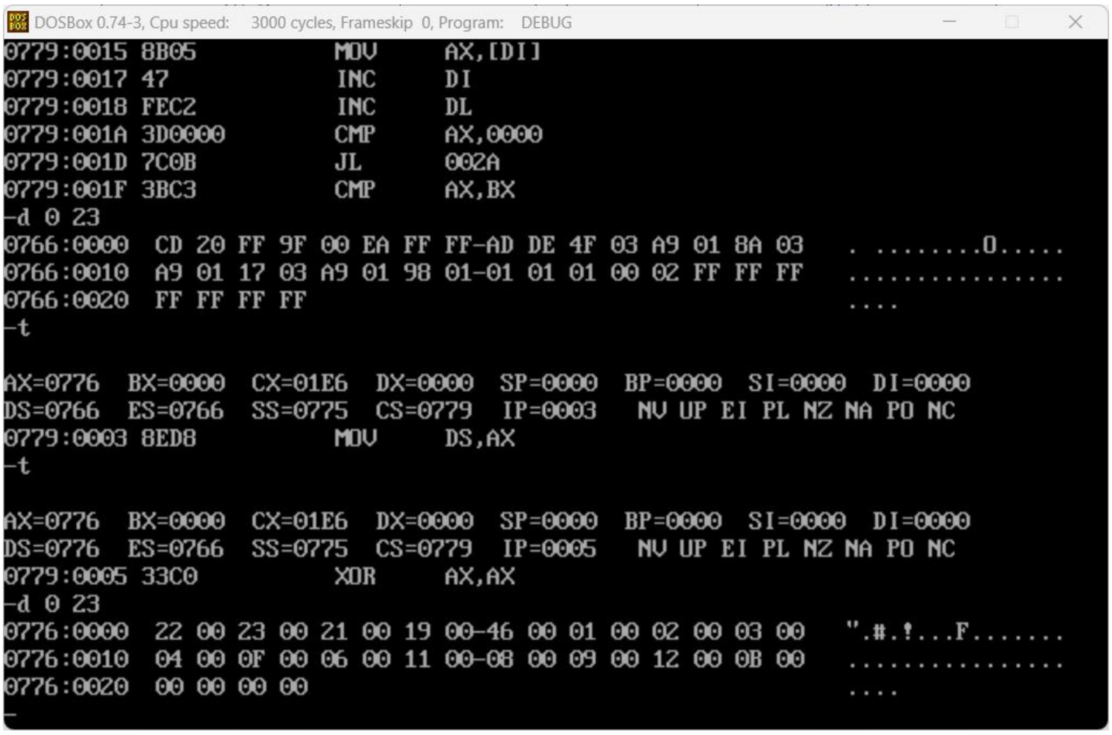
（1）反汇编的截图



DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

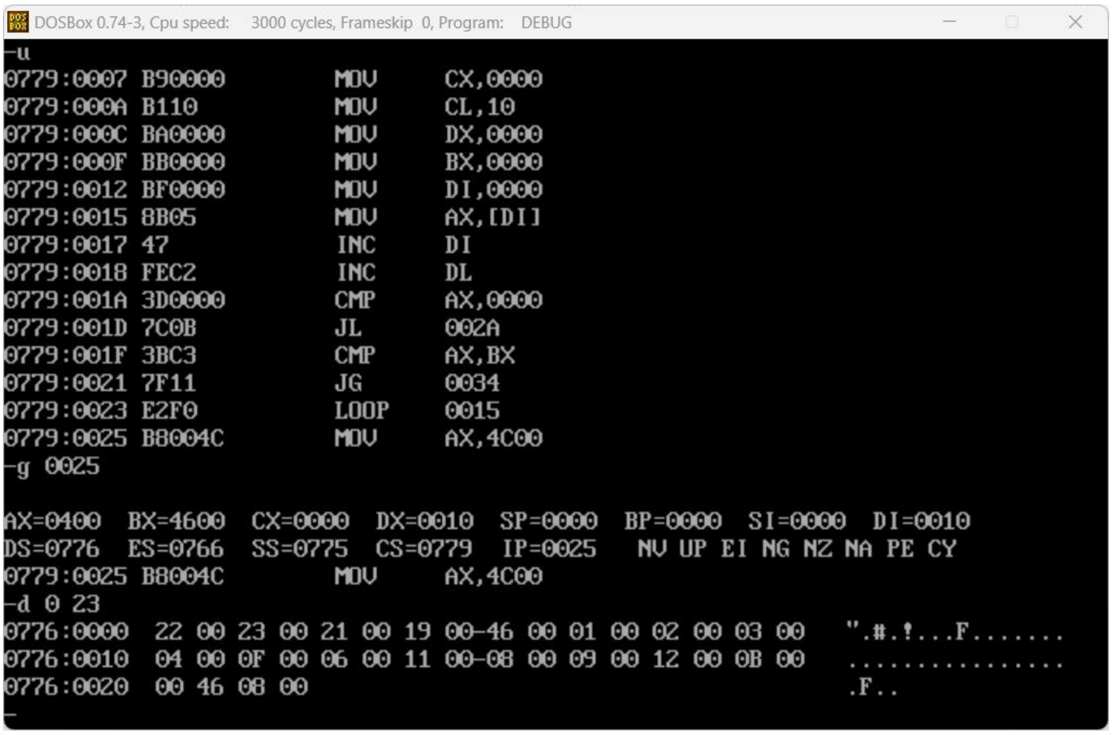
```
Z:\>set INIT=C:\MASM50

Z:\>set HELPFILES=C:\MASM50\BIN\*.HLP

Z:\>c:

C:\>cd learn

C:\LEARN>debug lab21.exe
-u
0779:0000 B87607        MOV     AX,0776
0779:0003 8ED8          MOV     DS,AX
0779:0005 33C0          XOR     AX,AX
0779:0007 B90000        MOV     CX,0000
0779:000A B110          MOV     CL,10
0779:000C BA0000        MOV     DX,0000
0779:000F BB0000        MOV     BX,0000
0779:0012 BF0000        MOV     DI,0000
0779:0015 8B05          MOV     AX,[DI]
0779:0017 47            INC     DI
0779:0018 FEC2          INC     DL
0779:001A 3D0000        CMP     AX,0000
0779:001D 7C0B          JL      002A
0779:001F 3BC3          CMP     AX,BX
-
```

（2）在进行计算前，显示数组 M 开始的 n+2 个字的内存值的截图（只能显示这 n+2 个字的内存值，多显示、少显示均扣分）


DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

```
0779:0015 8B05          MOV     AX,[DI]
0779:0017 47            INC     DI
0779:0018 FEC2          INC     DL
0779:001A 3D0000        CMP     AX,0000
0779:001D 7C0B          JL      002A
0779:001F 3BC3          CMP     AX,BX
-d 0 23
0766:0000  CD 20 FF 9F 00 EA FF FF-AD DE 4F 03 A9 01 8A 03   . ........O.....
0766:0010  A9 01 17 03 A9 01 98 01-01 01 01 00 02 FF FF FF   ................
0766:0020  FF FF FF FF                                       ....
-t

AX=0776  BX=0000  CX=01E6  DX=0000  SP=0000  BP=0000  SI=0000  DI=0000
DS=0766  ES=0766  SS=0775  CS=0779  IP=0003   NV UP EI PL NZ NA PO NC
0779:0003 8ED8          MOV     DS,AX
-t

AX=0776  BX=0000  CX=01E6  DX=0000  SP=0000  BP=0000  SI=0000  DI=0000
DS=0776  ES=0766  SS=0775  CS=0779  IP=0005   NV UP EI PL NZ NA PO NC
0779:0005 33C0          XOR     AX,AX
-d 0 23
0776:0000  22 00 23 00 21 00 19 00-46 00 01 00 02 00 03 00   ".#.!...F.......
0776:0010  04 00 0F 00 06 00 11 00-08 00 09 00 12 00 0B 00   ................
0776:0020  00 00 00 00                                       ....
-
```

（3）执行完计算后，显示数组 M 开始的 n+2 个字的内存值的截图（只能显示这 n+2 个字的内存值，多显示、少显示均扣分）


DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

```
-u
0779:0007 B90000        MOV     CX,0000
0779:000A B110          MOV     CL,10
0779:000C BA0000        MOV     DX,0000
0779:000F BB0000        MOV     BX,0000
0779:0012 BF0000        MOV     DI,0000
0779:0015 8B05          MOV     AX,[DI]
0779:0017 47            INC     DI
0779:0018 FEC2          INC     DL
0779:001A 3D0000        CMP     AX,0000
0779:001D 7C0B          JL      002A
0779:001F 3BC3          CMP     AX,BX
0779:0021 7F11          JG      0034
0779:0023 E2F0          LOOP    0015
0779:0025 B8004C        MOV     AX,4C00
-g 0025

AX=0400  BX=4600  CX=0000  DX=0010  SP=0000  BP=0000  SI=0000  DI=0010
DS=0776  ES=0766  SS=0775  CS=0779  IP=0025   NV UP EI NG NZ NA PE CY
0779:0025 B8004C        MOV     AX,4C00
-d 0 23
0776:0000  22 00 23 00 21 00 19 00-46 00 01 00 02 00 03 00   ".#.!...F.......
0776:0010  04 00 0F 00 06 00 11 00-08 00 09 00 12 00 0B 00   ................
0776:0020  00 46 08 00                                       .F..
-
```

（4）源代码

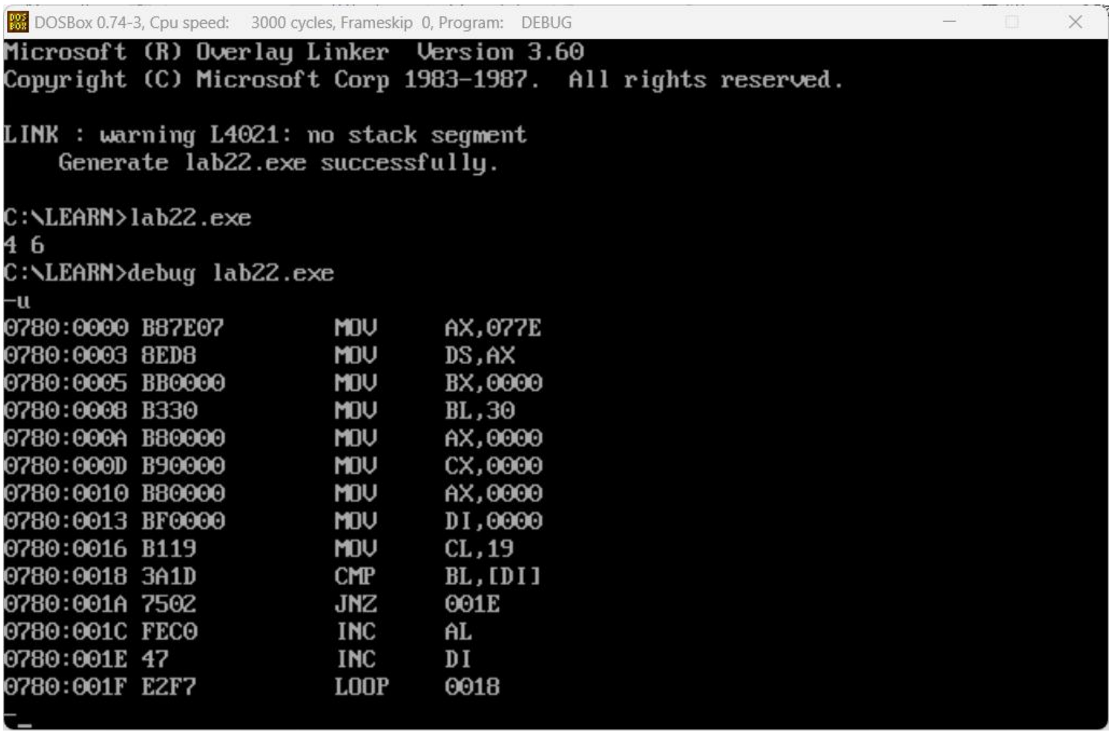
```
        data segment
      n equ 16
```

```
    M dw 22H,23H,21H,19H,46H,1,2,3,4,15,6,17,8,9,18,11
    RESULT dw 0
    P dw 0
data ends

code segment
    assume cs:code, ds:data
    main    proc
        ; assign the data segment base address to DS
        mov   ax, data
        mov   ds, ax

        xor   ax, ax
        mov   cx, 0
        mov   cl, n
        mov   dx, 0
        mov   bx, 0
        mov   di, offset M

    loop1:
        mov   ax, [di]
        inc   di
        inc   dl
        cmp   ax, 0
        jl    c2
        cmp   ax, bx
        jg    refresh1
    rett:
        loop  loop1

        mov   ax, 4c00h
        int   21h

    c2:
        neg   ax
        cmp   ax, bx
        jg    refresh2
        neg   ax
        jmp   rett

    refresh1:
        mov   bx,ax
        mov   [M+2*(n+1)],dl
        mov   [M+2*n],bx
```

```
            jmp    rett

        refresh2:
            neg    ax
            mov    bx,ax
            neg    ax
            mov    [M+2*(n+1)],dl
            mov    [M+2*n],bx
            jmp    rett


        main     endp
code ends
end main
```

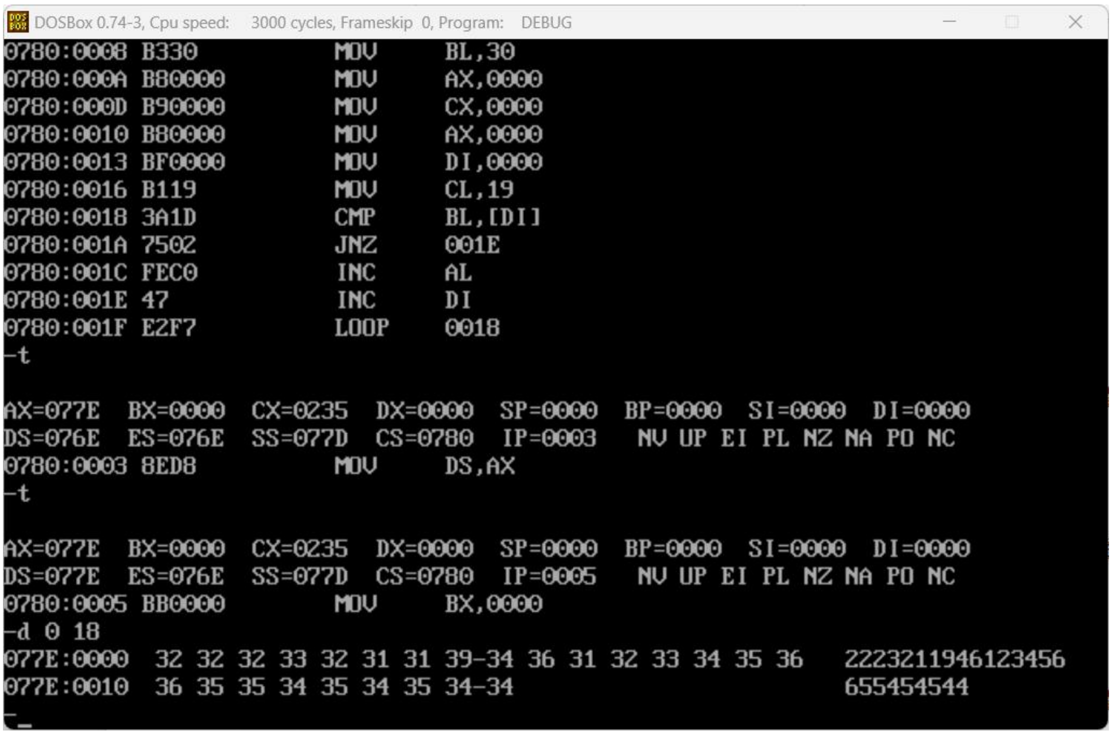
2、 分支程序设计

（1）反汇编的截图


DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

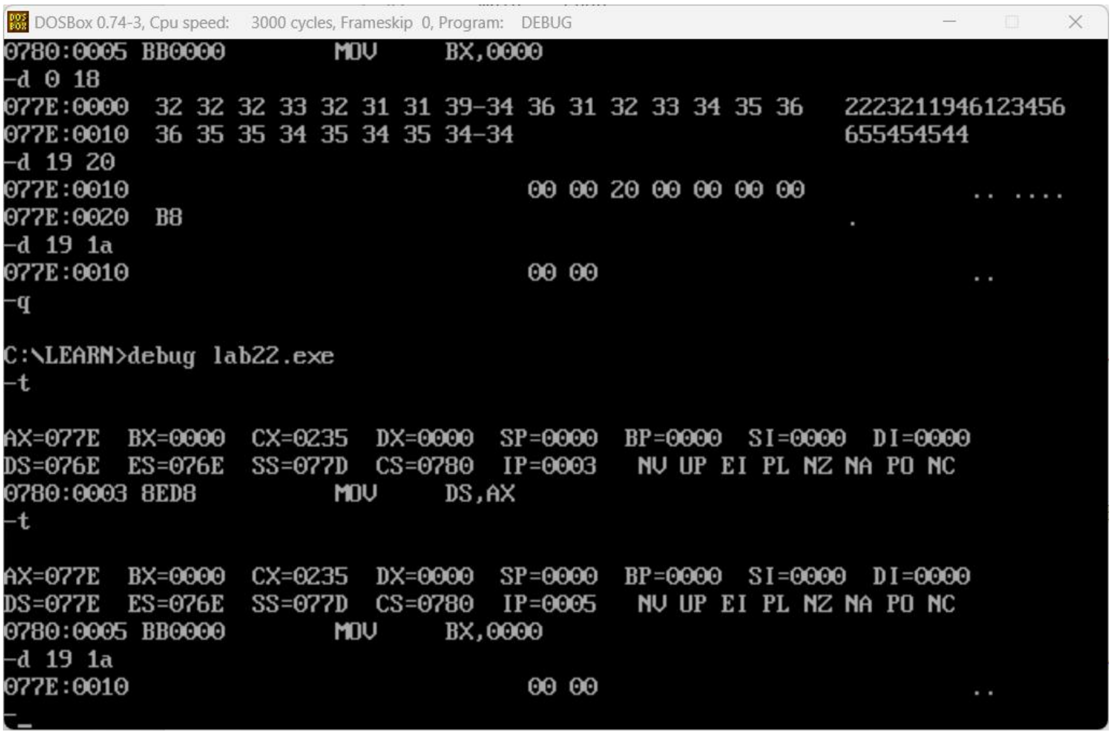
```
Microsoft (R) Overlay Linker  Version 3.60
Copyright (C) Microsoft Corp 1983-1987.  All rights reserved.

LINK : warning L4021: no stack segment
    Generate lab22.exe successfully.

C:\LEARN>lab22.exe
4 6
C:\LEARN>debug lab22.exe
-u
0780:0000 B87E07        MOV     AX,077E
0780:0003 8ED8          MOV     DS,AX
0780:0005 BB0000        MOV     BX,0000
0780:0008 B330          MOV     BL,30
0780:000A B80000        MOV     AX,0000
0780:000D B90000        MOV     CX,0000
0780:0010 B80000        MOV     AX,0000
0780:0013 BF0000        MOV     DI,0000
0780:0016 B119          MOV     CL,19
0780:0018 3A1D          CMP     BL,[DI]
0780:001A 7502          JNZ     001E
0780:001C FEC0          INC     AL
0780:001E 47            INC     DI
0780:001F E2F7          LOOP    0018
-_
```

（2）在进行计算前，显示在数据段中定义的含学号的字符串的内存值的截图（只能显示该完整的字符串，多显示、少显示均扣分）


DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

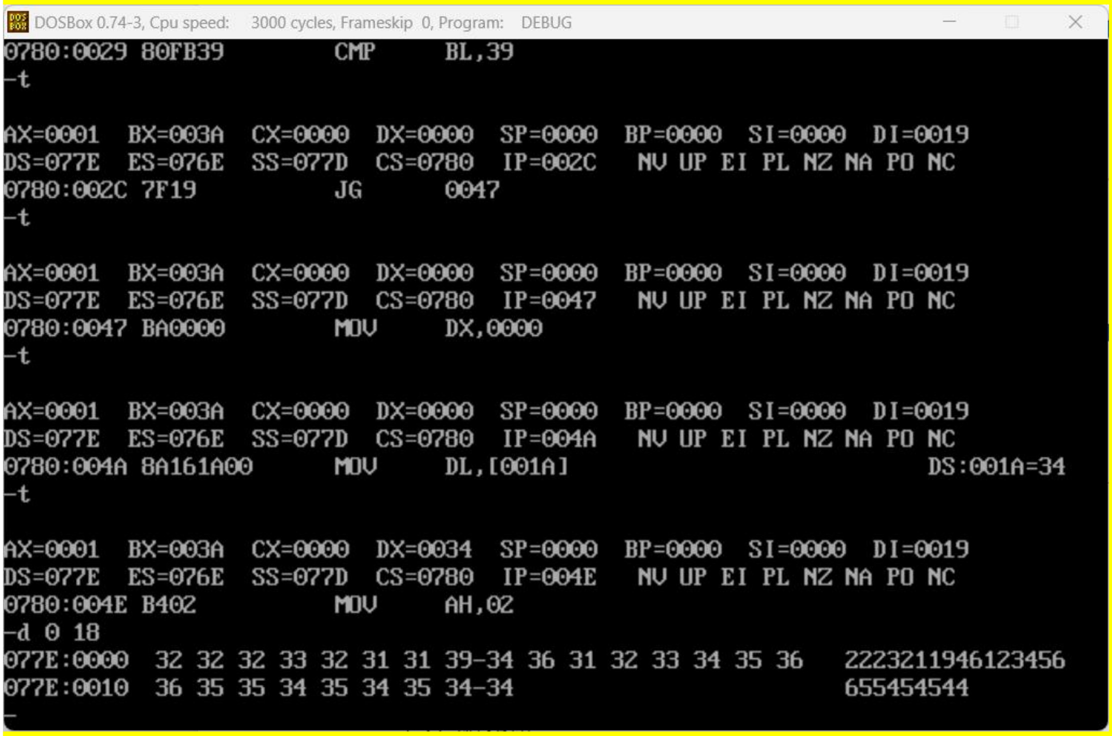
```
0780:0008 B330          MOV     BL,30
0780:000A B80000        MOV     AX,0000
0780:000D B90000        MOV     CX,0000
0780:0010 B80000        MOV     AX,0000
0780:0013 BF0000        MOV     DI,0000
0780:0016 B119          MOV     CL,19
0780:0018 3A1D          CMP     BL,[DI]
0780:001A 7502          JNZ     001E
0780:001C FEC0          INC     AL
0780:001E 47            INC     DI
0780:001F E2F7          LOOP    0018
-t

AX=077E  BX=0000  CX=0235  DX=0000  SP=0000  BP=0000  SI=0000  DI=0000
DS=076E  ES=076E  SS=077D  CS=0780  IP=0003   NV UP EI PL NZ NA PO NC
0780:0003 8ED8          MOV     DS,AX
-t

AX=077E  BX=0000  CX=0235  DX=0000  SP=0000  BP=0000  SI=0000  DI=0000
DS=077E  ES=076E  SS=077D  CS=0780  IP=0005   NV UP EI PL NZ NA PO NC
0780:0005 BB0000        MOV     BX,0000
-d 0 18
077E:0000  32 32 32 33 32 31 31 39-34 36 31 32 33 34 35 36   2223211946123456
077E:0010  36 35 35 34 35 34 35 34-34                        655454544
-_
```

（3）在进行计算前，显示在数据段中定义的 COUNT 数组的内存值的截图（只能显示完整的 COUNT 数组内容，多显示、少显示均扣分）


DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

```
0780:0005 BB0000        MOV     BX,0000
-d 0 18
077E:0000  32 32 32 33 32 31 31 39-34 36 31 32 33 34 35 36   2223211946123456
077E:0010  36 35 35 34 35 34 35 34-34                        655454544
-d 19 20
077E:0010                             00 00 20 00 00 00 00            .. ....
077E:0020  B8                                                .
-d 19 1a
077E:0010                             00 00                           ..
-q

C:\LEARN>debug lab22.exe
-t

AX=077E  BX=0000  CX=0235  DX=0000  SP=0000  BP=0000  SI=0000  DI=0000
DS=076E  ES=076E  SS=077D  CS=0780  IP=0003   NV UP EI PL NZ NA PO NC
0780:0003 8ED8          MOV     DS,AX
-t

AX=077E  BX=0000  CX=0235  DX=0000  SP=0000  BP=0000  SI=0000  DI=0000
DS=077E  ES=076E  SS=077D  CS=0780  IP=0005   NV UP EI PL NZ NA PO NC
0780:0005 BB0000        MOV     BX,0000
-d 19 1a
077E:0010                             00 00                           ..
-_
```

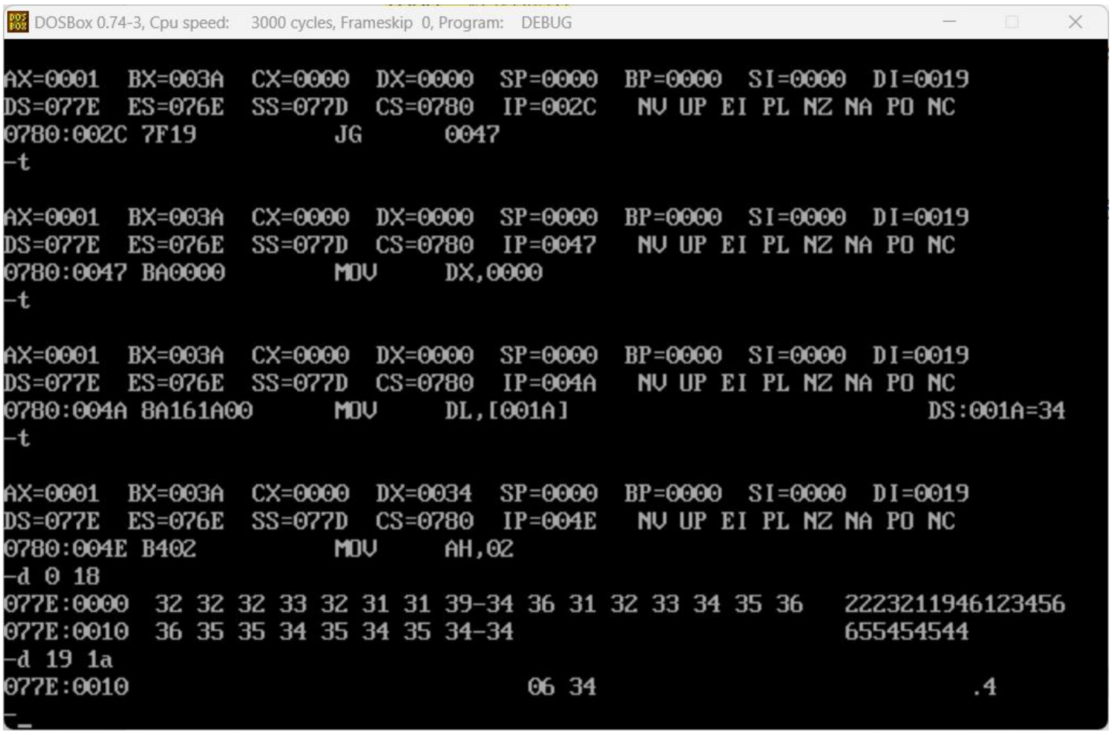
（4）执行完计算后，显示在数据段中定义的含学号的字符串的内存值的截图（只能显示该完整的字符串，多显示、少显示均扣分）


DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

```
0780:0029 80FB39        CMP     BL,39
-t

AX=0001  BX=003A  CX=0000  DX=0000  SP=0000  BP=0000  SI=0000  DI=0019
DS=077E  ES=076E  SS=077D  CS=0780  IP=002C   NV UP EI PL NZ NA PO NC
0780:002C 7F19          JG      0047
-t

AX=0001  BX=003A  CX=0000  DX=0000  SP=0000  BP=0000  SI=0000  DI=0019
DS=077E  ES=076E  SS=077D  CS=0780  IP=0047   NV UP EI PL NZ NA PO NC
0780:0047 BA0000        MOV     DX,0000
-t

AX=0001  BX=003A  CX=0000  DX=0000  SP=0000  BP=0000  SI=0000  DI=0019
DS=077E  ES=076E  SS=077D  CS=0780  IP=004A   NV UP EI PL NZ NA PO NC
0780:004A 8A161A00      MOV     DL,[001A]                          DS:001A=34
-t

AX=0001  BX=003A  CX=0000  DX=0034  SP=0000  BP=0000  SI=0000  DI=0019
DS=077E  ES=076E  SS=077D  CS=0780  IP=004E   NV UP EI PL NZ NA PO NC
0780:004E B402          MOV     AH,02
-d 0 18
077E:0000  32 32 32 33 32 31 31 39-34 36 31 32 33 34 35 36   2223211946123456
077E:0010  36 35 35 34 35 34 35 34-34                        655454544
-
```

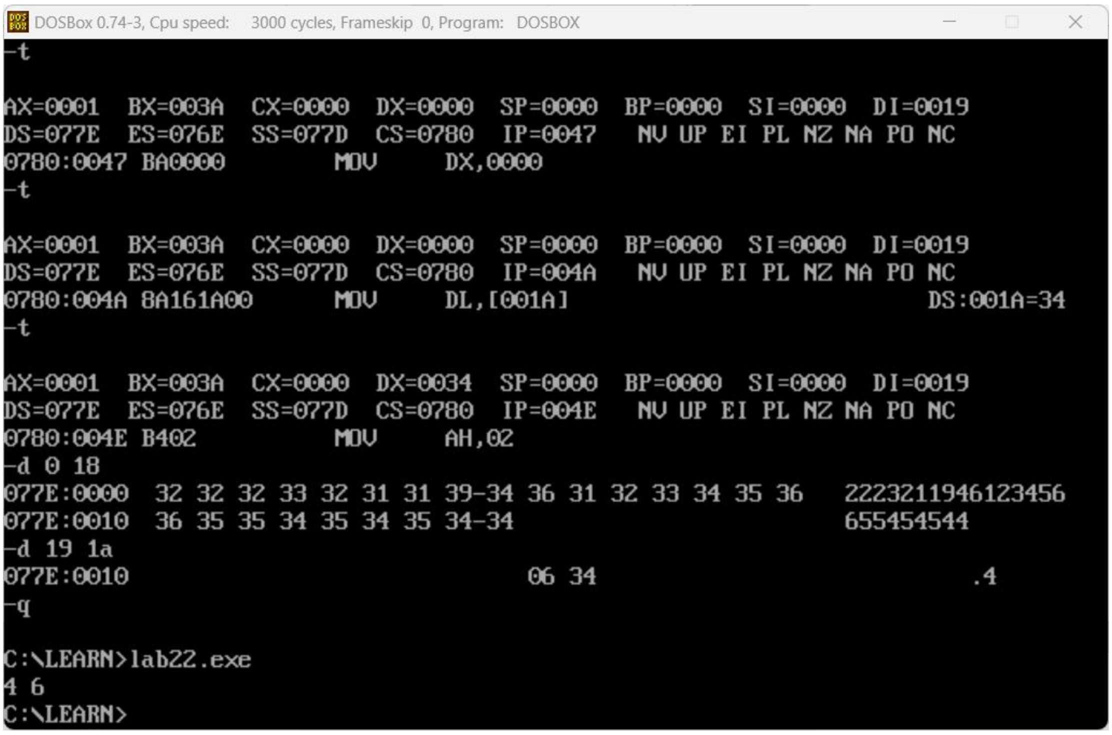
:

（5）执行完计算后，显示在数据段中定义的 COUNT 数组的内存值的截图（只能显示完整的 COUNT 数组内容，多显示、少显示均扣分）


DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

```
AX=0001  BX=003A  CX=0000  DX=0000  SP=0000  BP=0000  SI=0000  DI=0019
DS=077E  ES=076E  SS=077D  CS=0780  IP=002C   NV UP EI PL NZ NA PO NC
0780:002C 7F19          JG      0047
-t

AX=0001  BX=003A  CX=0000  DX=0000  SP=0000  BP=0000  SI=0000  DI=0019
DS=077E  ES=076E  SS=077D  CS=0780  IP=0047   NV UP EI PL NZ NA PO NC
0780:0047 BA0000        MOV     DX,0000
-t

AX=0001  BX=003A  CX=0000  DX=0000  SP=0000  BP=0000  SI=0000  DI=0019
DS=077E  ES=076E  SS=077D  CS=0780  IP=004A   NV UP EI PL NZ NA PO NC
0780:004A 8A161A00      MOV     DL,[001A]                          DS:001A=34
-t

AX=0001  BX=003A  CX=0000  DX=0034  SP=0000  BP=0000  SI=0000  DI=0019
DS=077E  ES=076E  SS=077D  CS=0780  IP=004E   NV UP EI PL NZ NA PO NC
0780:004E B402          MOV     AH,02
-d 0 18
077E:0000  32 32 32 33 32 31 31 39-34 36 31 32 33 34 35 36   2223211946123456
077E:0010  36 35 35 34 35 34 35 34-34                        655454544
-d 19 1a
077E:0010                              06 34                          .4
-_
```

（6）程序在 DOSBox 下直接运行的截图



```
-t

AX=0001  BX=003A  CX=0000  DX=0000  SP=0000  BP=0000  SI=0000  DI=0019
DS=077E  ES=076E  SS=077D  CS=0780  IP=0047   NV UP EI PL NZ NA PO NC
0780:0047 BA0000        MOV     DX,0000
-t

AX=0001  BX=003A  CX=0000  DX=0000  SP=0000  BP=0000  SI=0000  DI=0019
DS=077E  ES=076E  SS=077D  CS=0780  IP=004A   NV UP EI PL NZ NA PO NC
0780:004A 8A161A00      MOV     DL,[001A]                          DS:001A=34
-t

AX=0001  BX=003A  CX=0000  DX=0034  SP=0000  BP=0000  SI=0000  DI=0019
DS=077E  ES=076E  SS=077D  CS=0780  IP=004E   NV UP EI PL NZ NA PO NC
0780:004E B402          MOV     AH,02
-d 0 18
077E:0000  32 32 32 33 32 31 31 39-34 36 31 32 33 34 35 36   2223211946123456
077E:0010  36 35 35 34 35 34 35 34-34                        655454544
-d 19 1a
077E:0010                             06 34                           .4
-q

C:\LEARN>lab22.exe
4 6
C:\LEARN>
```

（7）源代码

```
    data ends

code segment
assume cs:code, ds:data

start:  mov ax,data
        mov ds,ax
        mov bx,0  ;
        mov bl,30h;
        mov ax,0  ;计数
        mov cx,0
    lp09:
        mov ax,0
        mov di,offset message
        mov cl,25
        loopStr: cmp bl,[di]
                 jne j1
                 inc al
              j1:inc di
                 loop loopStr ;此时 bl 存的数，al 存的个数
        cmp al,count
        jge refresh     ;如果个数大于等于就更新
    j3: inc bl
        cmp bl,39h
        jg  j4
        jle  lp09

        refresh:cmp al,count
            jne j2
            cmp bl,[count+1]
            jg  j2
            jle  j3

        j2: mov [count+1],bl
            mov count,al
            jmp j3


        j4:
        mov dx,0
        mov dl, [count+1]
        mov ah, 02h
        int 21h

        mov dx, 0
        mov ax, 0


        mov dl, [space]
        mov ah, 02h
        int 21h

        mov dx, 0
        mov ax, 0

        mov dl, [count]
        add dl, '0'
        mov ah, 02h
        int 21h



        mov   ax, 4c00h
        int   21h

code ends
end start
```