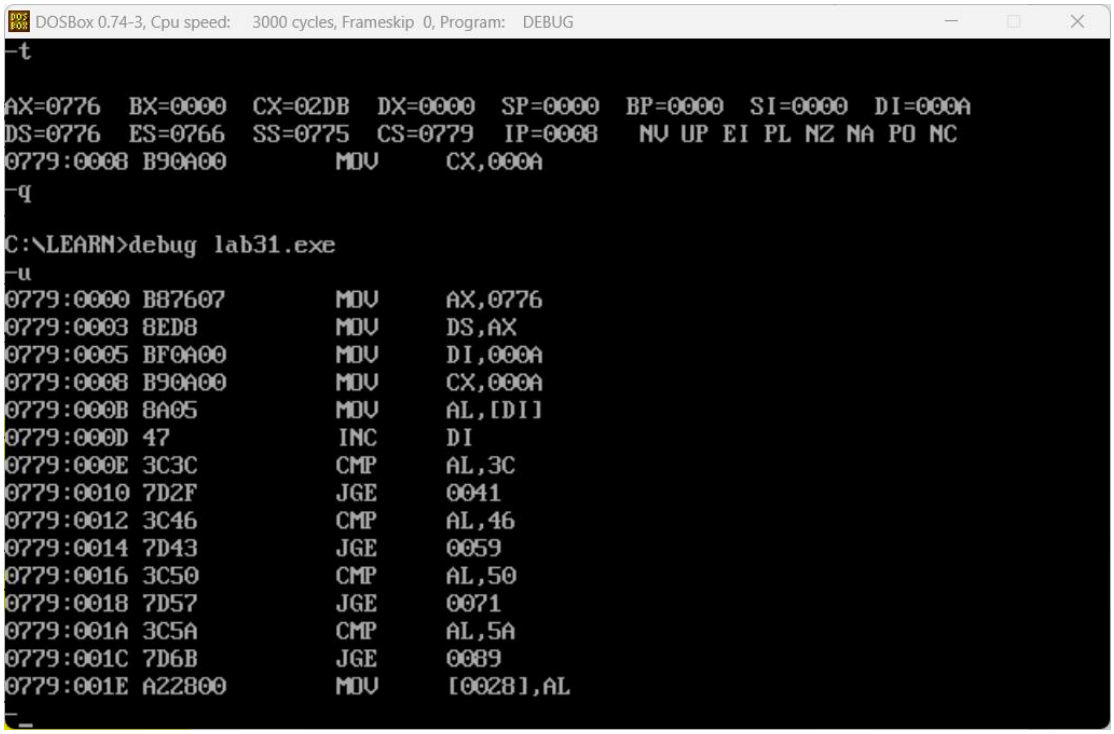
第 3 次上机(x86)

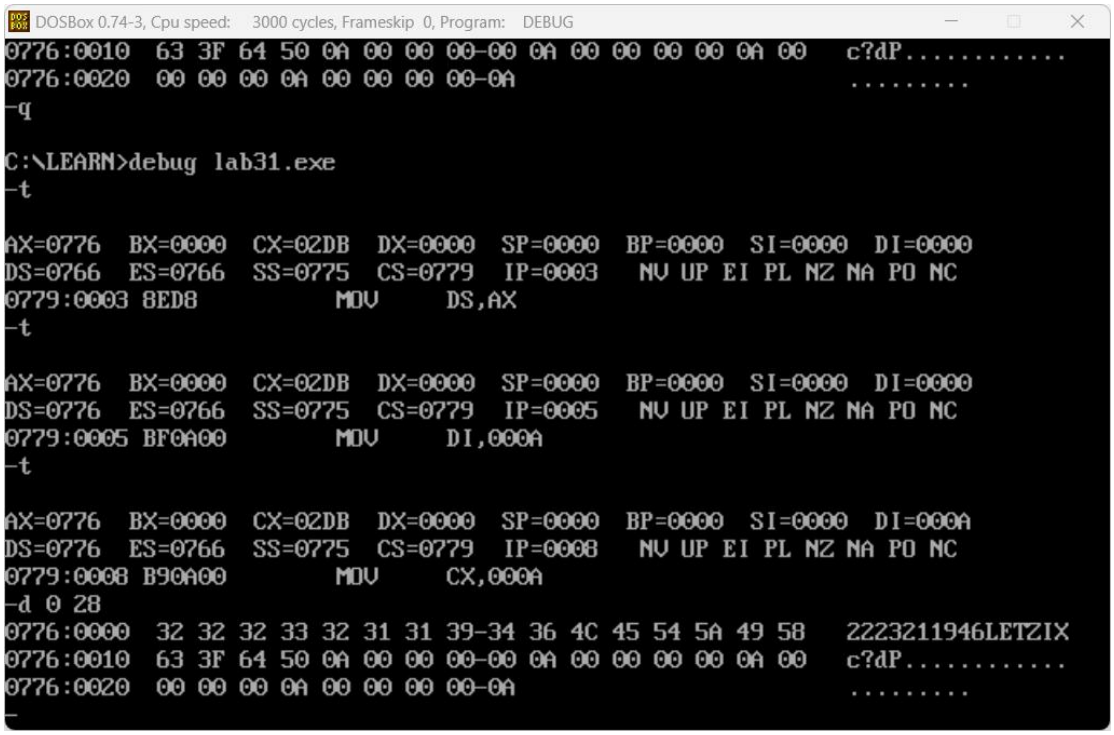
班级	学号	姓名
计试 2201	2223211946	王顺平

1、 子程序设计

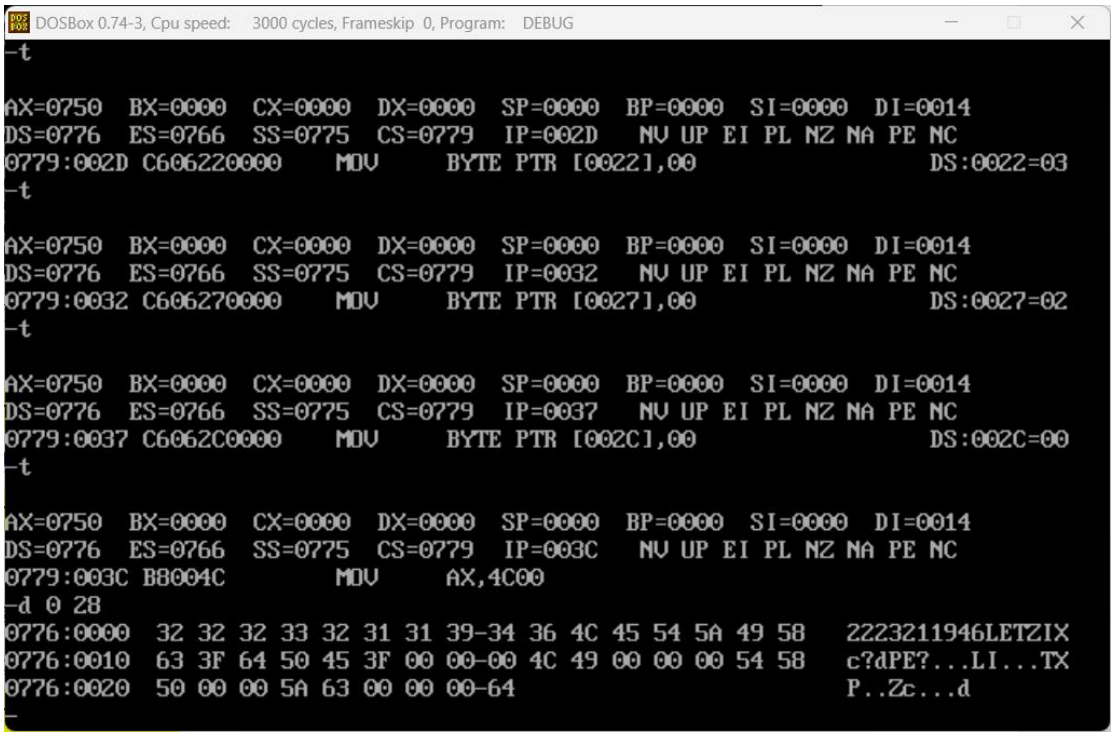
（1）反汇编的截图



（2）在进行计算前，显示 ID、array 以及 S6——S10 的内存值的截图（多显示、少显示均扣分）



（3）执行完计算后，显示 ID、array 以及 S6——S10 的内存值的截图（多显示、少显示均扣分）



（4）源代码

```
1.        data segment
2.       ID  db  '2223211946'
```

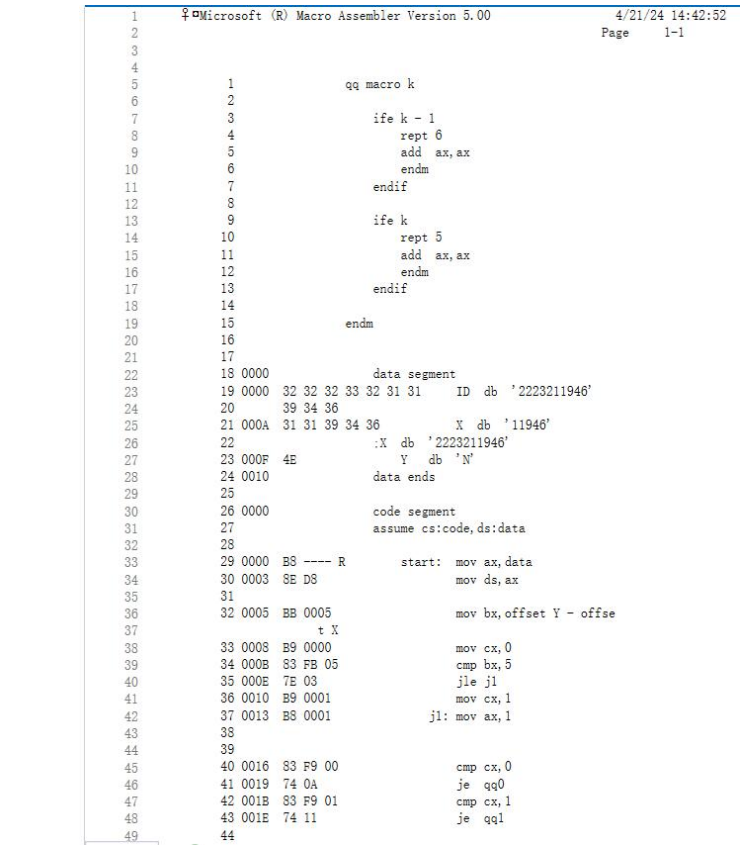
```
    array db 76,69,84,90,73,88,99,63,100,80
    S6  db  10,0,0,0,0
    S7  db  10,0,0,0,0
    S8  db  10,0,0,0,0
    S9  db  10,0,0,0,0
    S10  db  10
data ends

code segment
    assume cs:code,ds:data

start:  mov ax,data
        mov ds,ax

        mov di,offset array
        mov cx,10
loop1:  mov al,[di]
        inc di
        cmp al,60
        jge ss6
j7:     cmp al,70
        jge ss7
j8:     cmp al,80
        jge ss8
j9:     cmp al,90
        jge ss9
j10:    mov [S10],al
retL:   loop loop1

        mov [S6 + 4],0
        mov [S7 + 4],0
        mov [S8 + 4],0
        mov [S9 + 4],0
        mov [S10 + 4],0

        mov   ax, 4c00h
        int   21h

ss6:    cmp al,70
        jge  j7

        push di
        mov di,word ptr[S6 + 4]
        and di,00FFh
```

```
47.           mov [S6 + di],al
48.           inc [S6 + 4]
49.           pop di
50.
51.           jmp retL
52.
53.   ss7:    cmp al,80
54.           jge  j8
55.
56.           push di
57.           mov di,word ptr[S7 + 4]
58.           and di,00FFh
59.           mov [S7 + di],al
60.           inc [S7 + 4]
61.           pop di
62.
63.           jmp retL
64.
65.   ss8:    cmp al,90
66.           jge  j9
67.
68.           push di
69.           mov di,word ptr[S8 + 4]
70.           and di,00FFh
71.           mov [S8 + di],al
72.           inc [S8 + 4]
73.           pop di
74.
75.           jmp retL
76.
77.   ss9:    cmp al,100
78.           jge  j10
79.
80.           push di
81.           mov di,word ptr[S9 + 4]
82.           and di,00FFh
83.           mov [S9 + di],al
84.           inc [S9 + 4]
85.           pop di
86.
87.           jmp retL
88.
89.
90.   code ends
```

```
91.  end start
```

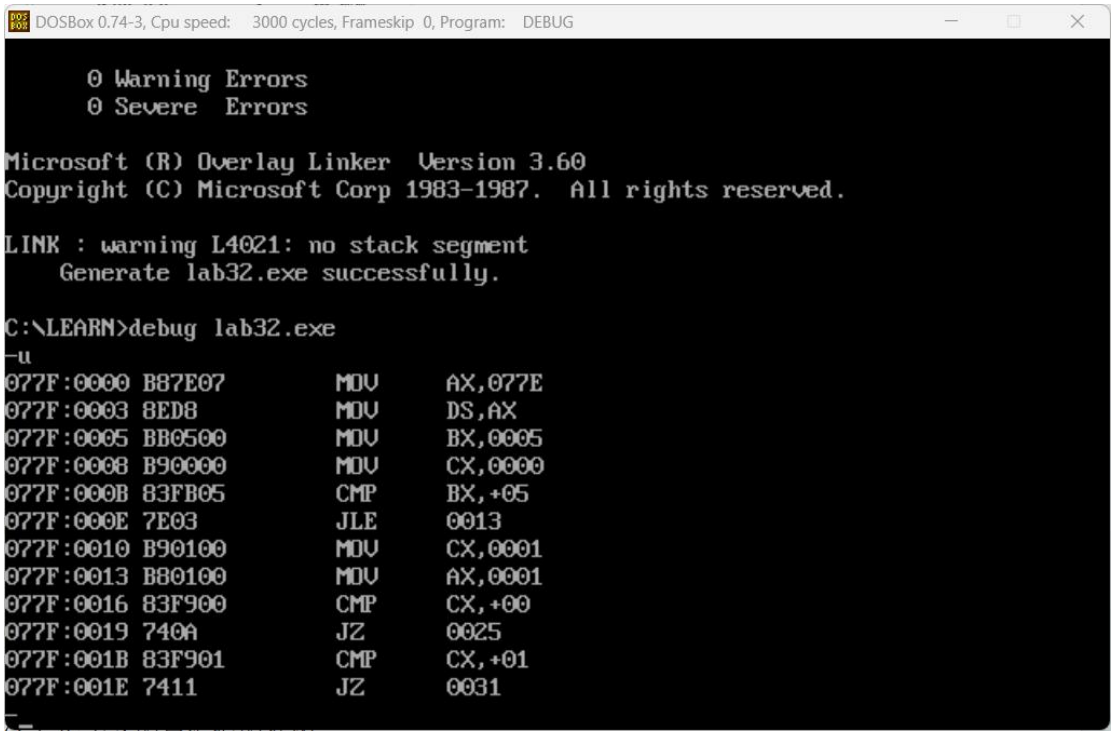
2、 高级汇编语言技术

（1）场景 1 的.lst 文件的截图

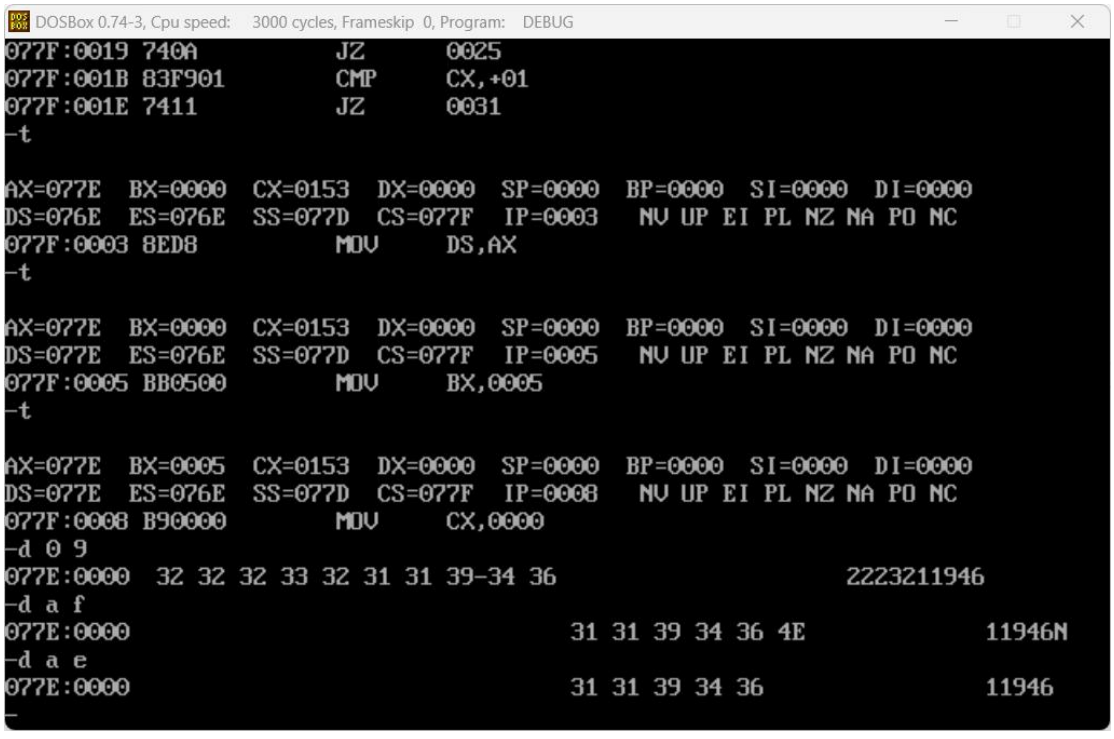


```
♀¤Microsoft (R) Macro Assembler Version 5.00                4/21/24 14:42:52
                                                            Page     1-1


                        qq macro k

                            ife k - 1
                                rept 6
                                add  ax,ax
                                endm
                            endif

                            ife k
                                rept 5
                                add  ax,ax
                                endm
                            endif

                        endm


0000                    data segment
0000  32 32 32 33 32 31 31      ID  db  '2223211946'
      39 34 36
000A  31 31 39 34 36            X  db  '11946'
                                :X  db  '2223211946'
000F  4E                        Y   db  'N'
0010                    data ends

0000                    code segment
                        assume cs:code,ds:data

0000  B8 ---- R         start:  mov ax,data
0003  8E D8                     mov ds,ax

0005  BB 0005                   mov bx,offset Y - offse
                                  t X
0008  B9 0000                   mov cx,0
000B  83 FB 05                  cmp bx,5
000E  7E 03                     jle j1
0010  B9 0001                   mov cx,1
0013  B8 0001                j1: mov ax,1

0016  83 F9 00                  cmp cx,0
0019  74 0A                     je  qq0
001B  83 F9 01                  cmp cx,1
001E  74 11                     je  qq1

0020  B8 4C00             r1: mov ax,4c00h
0023  CD 21                   int 21h

0025                    qq0:
                        qq 0
0025  03 C0                 2         add  ax,ax
0027  03 C0                 2         add  ax,ax
0029  03 C0                 2         add  ax,ax
002B  03 C0                 2         add  ax,ax
♀¤Microsoft (R) Macro Assembler Version 5.00                4/
                                                            Page


002D  03 C0                 2         add  ax,ax
002F  EB EF                   jmp r1
0031                    qq1:
                        qq 1
0031  03 C0                 2         add  ax,ax
0033  03 C0                 2         add  ax,ax
0035  03 C0                 2         add  ax,ax
0037  03 C0                 2         add  ax,ax
0039  03 C0                 2         add  ax,ax
003B  03 C0                 2         add  ax,ax
003D  EB E1                   jmp r1

003F                code ends
              end start

♀¤Microsoft (R) Macro Assembler Version 5.00                4/
                                                            Symb


Macros:

        N a m e         Lines

QQ . . . . . . . . . . . . . .      10

Segments and Groups:

                N a m e             Length   Align  Combine Class

CODE . . . . . . . . . . . . .      003F    PARA    NONE
DATA . . . . . . . . . . . . .      0010    PARA    NONE

Symbols:

                N a m e             Type     Value   Attr


                                                    --
ID . . . . . . . . . . . . . .      L BYTE  0000    DATA

J1 . . . . . . . . . . . . . .      L NEAR  0013    CODE

QQ0 . . . . . . . . . . . . .       L NEAR  0025    CODE
QQ1 . . . . . . . . . . . . .       L NEAR  0031    CODE

RL . . . . . . . . . . . . . .      L NEAR  0020    CODE

START . . . . . . . . . . . .       L NEAR  0000    CODE

X . . . . . . . . . . . . . .       L BYTE  000A    DATA

Y . . . . . . . . . . . . . .       L BYTE  000F    DATA

@FILENAME . . . . . . . . . .       TEXT  lab32


     55 Source  Lines
     86 Total   Lines
     17 Symbols

  50386 + 465998 Bytes symbol space free

      0 Warning Errors
      0 Severe  Errors
```

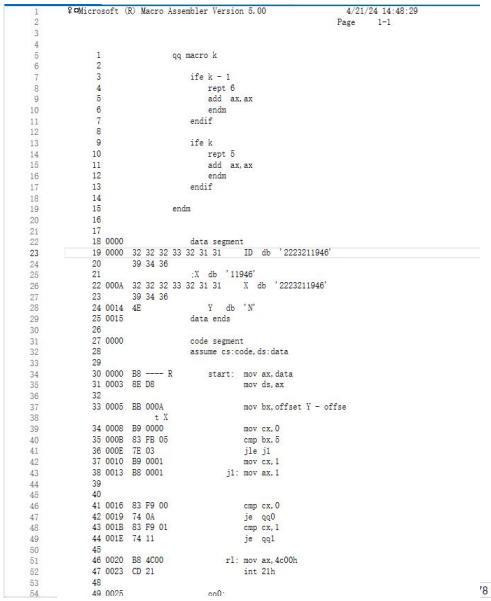
（2）场景 1 的反汇编的截图



（3）场景 1 的显示 X 的内存值的截图（多显示、少显示均扣分）

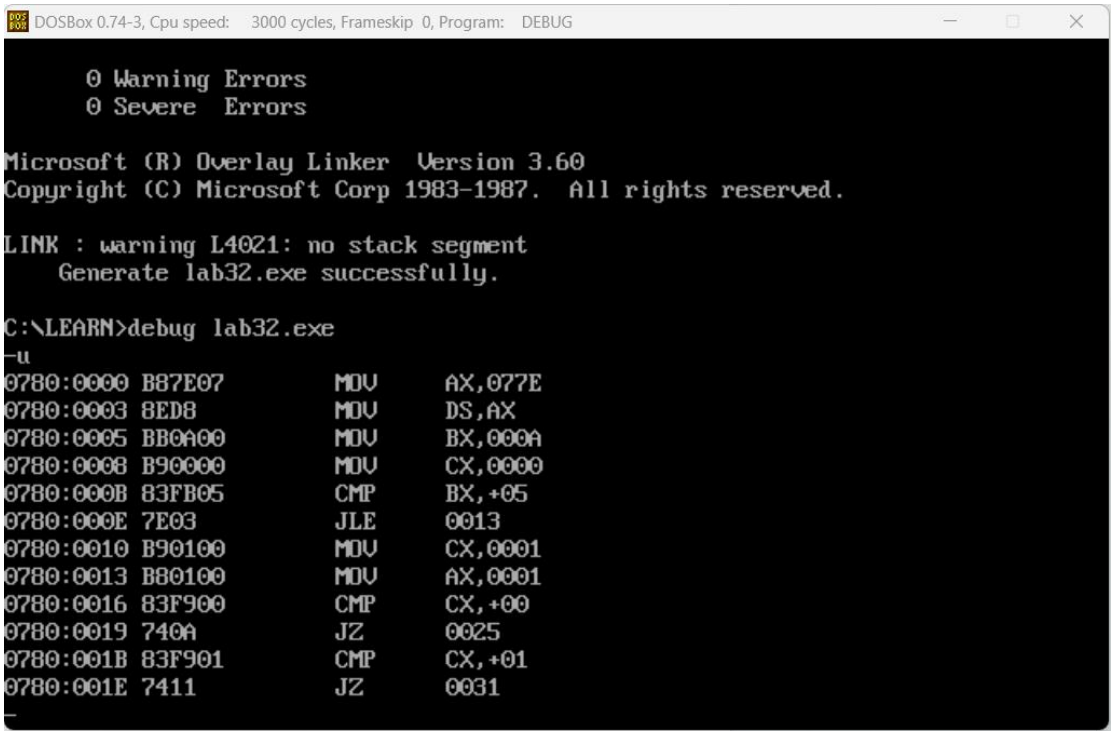


（4）场景 2 的.lst 文件的截图

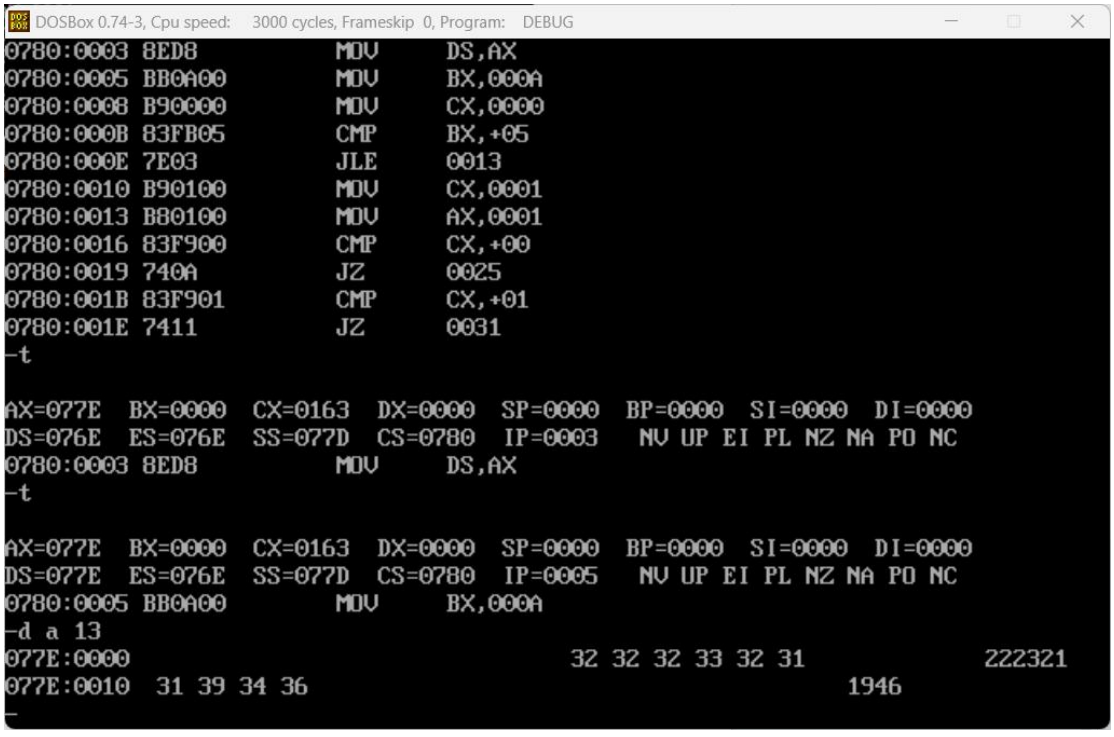


```
Microsoft (R) Macro Assembler Version 5.00                  4/21/24 14:48:29
                                                            Page     1-1


                qq macro k
                    ife k - 1
                        rept 6
                        add  ax,ax
                        endm
                    endif
                    ife k
                        rept 5
                        add  ax,ax
                        endm
                    endif
                endm
0000            data segment
0000 32 32 32 33 32 31 31       ID  db  '2223211946'
     39 34 36
                                :X  db  '11946'
000A 32 32 32 33 32 31 31       X  db  '2223211946'
     39 34 36
0014 4E                         Y   db  'N'
0015            data ends

0000            code segment
                assume cs:code,ds:data

0000 B8 ---- R      start:  mov ax,data
0003 8E D8                  mov ds,ax

0005 BB 000A                mov bx,offset Y - offse
                    t X
0008 B9 0000                mov cx,0
000B 83 FB 05               cmp bx,5
000E 7E 03                  jle j1
0010 B9 0001                mov cx,1
0013 B8 0001            j1: mov ax,1

0016 83 F9 00               cmp cx,0
0019 74 0A                  je  qq0
001B 83 F9 01               cmp cx,1
001E 74 11                  je  qq1

0020 B8 4C00            r1: mov ax,4c00h
0023 CD 21                  int 21h

0025            qq0:
                    50              qq 0
0025 03 C0                  2       add  ax,ax
0027 03 C0                  2       add  ax,ax
0029 03 C0                  2       add  ax,ax
Microsoft (R) Macro Assembler Version 5.00                  4/21/24 14:48:29
                                                            Page     1-2


002B 03 C0                  2       add  ax,ax
002D 03 C0                  2       add  ax,ax
002F EB EF                      jmp r1
0031                    qq1:
                        qq 1
0031 03 C0                  2       add  ax,ax
0033 03 C0                  2       add  ax,ax
0035 03 C0                  2       add  ax,ax
0037 03 C0                  2       add  ax,ax
0039 03 C0                  2       add  ax,ax
003B 03 C0                  2       add  ax,ax
003D EB E1                      jmp r1

003F                code ends
                end start
Microsoft (R) Macro Assembler Version 5.00                  4/21/24 14:48:29
                                                            Symbols-1


Macros:

        N a m e         Lines

QQ . . . . . . . . . . . . . .      10

Segments and Groups:

                N a m e             Length   Align  Combine Class

CODE . . . . . . . . . . . . .      003F    PARA    NONE
DATA . . . . . . . . . . . . .      0015    PARA    NONE

Symbols:

                N a m e             Type     Value   Attr

ID . . . . . . . . . . . . . .      L BYTE   0000    DATA

J1 . . . . . . . . . . . . . .      L NEAR   0013    CODE

QQ0 . . . . . . . . . . . . .       L NEAR   0025    CODE
QQ1 . . . . . . . . . . . . .       L NEAR   0031    CODE

RL . . . . . . . . . . . . . .      L NEAR   0020    CODE

START  . . . . . . . . . . . .      L NEAR   0000    CODE

X  . . . . . . . . . . . . . .      L BYTE   000A    DATA

Y  . . . . . . . . . . . . . .      L BYTE   0014    DATA

@FILENAME  . . . . . . . . . .      TEXT  lab32


     55 Source  Lines
     86 Total   Lines
     17 Symbols

  50386 + 465998 Bytes symbol space free

      0 Warning Errors
      0 Severe  Errors
```

（5）场景 2 的反汇编的截图



（6）场景 2 的显示 X 的内存值的截图（多显示、少显示均扣分）



（7）源代码

```
1.      qq macro k
2.
3.      ife k - 1
4.          rept 6
```

```
            add  ax,ax
            endm
        endif

        ife k
            rept 5
            add  ax,ax
            endm
        endif

endm


data segment
    ID  db  '2223211946'
    ;X  db  '11946'
    X  db  '2223211946'
    Y   db  'N'
data ends

code segment
    assume cs:code,ds:data

start:  mov ax,data
        mov ds,ax

        mov bx,offset Y - offset X
        mov cx,0
        cmp bx,5
        jle j1
        mov cx,1
    j1: mov ax,1


        cmp cx,0
        je  qq0
        cmp cx,1
        je  qq1

    rl: mov ax,4c00h
        int 21h

    qq0:
        qq 0
```

```
        jmp rl
    qq1:
        qq 1
        jmp rl

code ends
end start
```