Name : Tazmeen Afroz
Roll No : 2pp-9252
COAL-Lab-Task-7

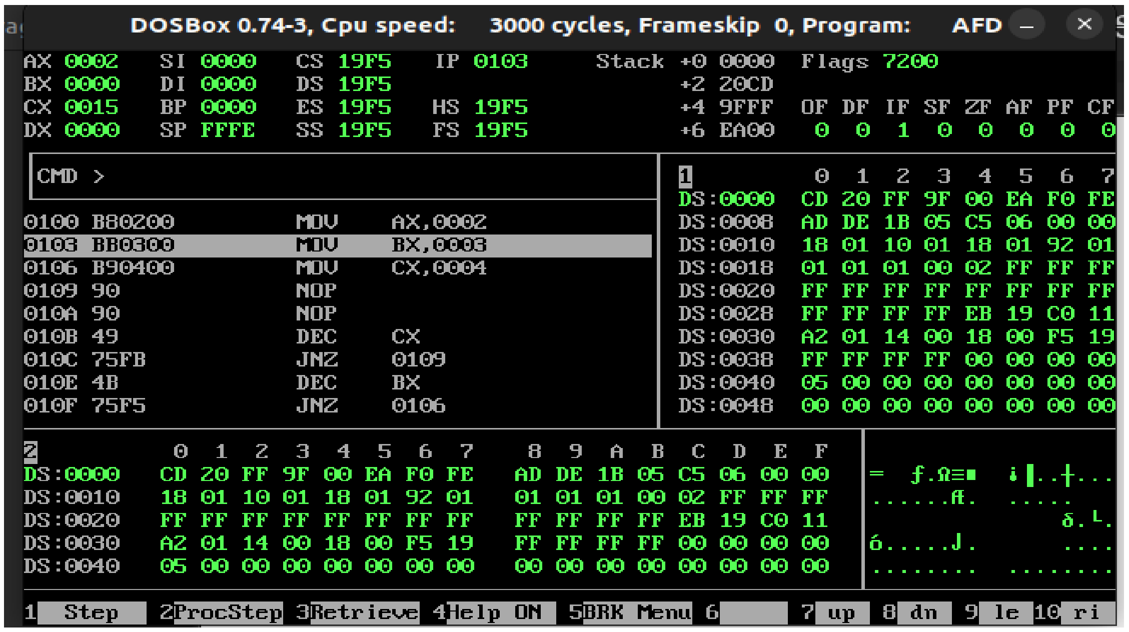
CODE

```
[org 0x0100]

   mov ax, 2
D1:
   mov bx, 3
D2:
   mov cx, 4
D3:
   NOP
   NOP
   DEC cx

 jne D3
 DEC bx
 jne D2
 DEC ax
 jne D1
 RET
```

Initialization

mov ax, 2

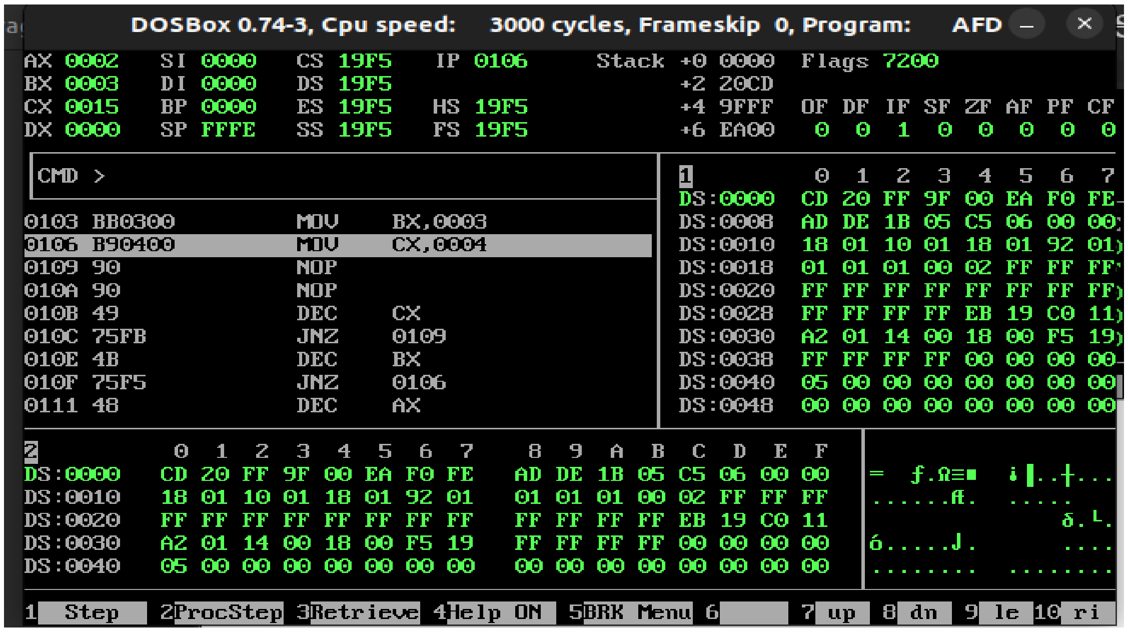
- Moves the value 2 into the AX register.
- AX now holds the value 2.



```
DOSBox 0.74-3, Cpu speed:    3000 cycles, Frameskip  0, Program:    AFD
AX 0002    SI 0000    CS 19F5    IP 0103      Stack +0 0000   Flags 7200
BX 0000    DI 0000    DS 19F5                       +2 20CD
CX 0015    BP 0000    ES 19F5    HS 19F5            +4 9FFF   OF DF IF SF ZF AF PF CF
DX 0000    SP FFFE    SS 19F5    FS 19F5            +6 EA00    0  0  1  0  0  0  0  0

 CMD >                                              1            0  1  2  3  4  5  6  7
                                                    DS:0000   CD 20 FF 9F 00 EA F0 FE
0100 B80200            MOV     AX,0002              DS:0008   AD DE 1B 05 C5 06 00 00
0103 BB0300            MOV     BX,0003              DS:0010   18 01 10 01 18 01 92 01
0106 B90400            MOV     CX,0004              DS:0018   01 01 01 00 02 FF FF FF
0109 90                NOP                          DS:0020   FF FF FF FF FF FF FF FF
010A 90                NOP                          DS:0028   FF FF FF FF EB 19 C0 11
010B 49                DEC     CX                   DS:0030   A2 01 14 00 18 00 F5 19
010C 75FB              JNZ     0109                 DS:0038   FF FF FF FF 00 00 00 00
010E 4B                DEC     BX                   DS:0040   05 00 00 00 00 00 00 00
010F 75F5              JNZ     0106                 DS:0048   00 00 00 00 00 00 00 00

2            0  1  2  3  4  5  6  7    8  9  A  B  C  D  E  F
DS:0000     CD 20 FF 9F 00 EA F0 FE   AD DE 1B 05 C5 06 00 00   =  ƒ.Ω≡■   ¡▌..┼...
DS:0010     18 01 10 01 18 01 92 01   01 01 01 00 02 FF FF FF   .......Æ.  .....
DS:0020     FF FF FF FF FF FF FF FF   FF FF FF FF EB 19 C0 11              δ.└.
DS:0030     A2 01 14 00 18 00 F5 19   FF FF FF FF 00 00 00 00   ó.....J.      ....
DS:0040     05 00 00 00 00 00 00 00   00 00 00 00 00 00 00 00   ........   ........

1 Step  2ProcStep 3Retrieve 4Help ON 5BRK Menu 6       7 up  8 dn  9 le 10 ri
```

mov bx, 3

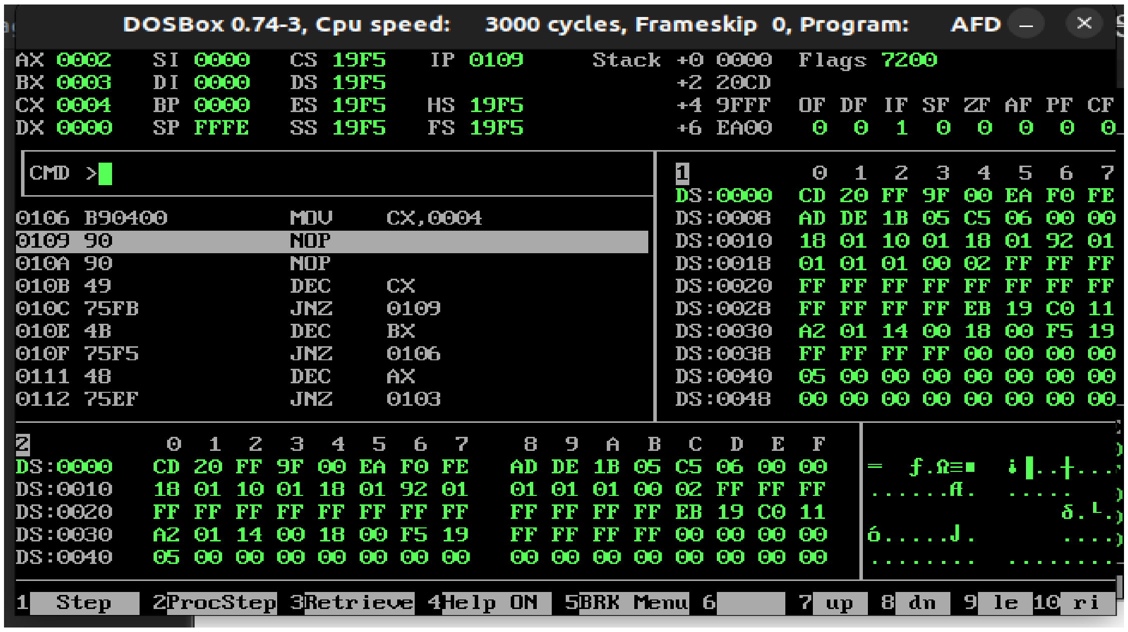
- Moves the value 3 into the BX register.
- BX now holds the value 3.



```
DOSBox 0.74-3, Cpu speed:    3000 cycles, Frameskip  0, Program:    AFD
AX 0002   SI 0000   CS 19F5   IP 0106      Stack +0 0000  Flags 7200
BX 0003   DI 0000   DS 19F5                      +2 20CD
CX 0015   BP 0000   ES 19F5   HS 19F5            +4 9FFF  OF DF IF SF ZF AF PF CF
DX 0000   SP FFFE   SS 19F5   FS 19F5            +6 EA00   0  0  1  0  0  0  0  0

 CMD >                                           1         0  1  2  3  4  5  6  7
                                                 DS:0000  CD 20 FF 9F 00 EA F0 FE
0103 BB0300          MOV     BX,0003             DS:0008  AD DE 1B 05 C5 06 00 00
0106 B90400          MOV     CX,0004             DS:0010  18 01 10 01 18 01 92 01
0109 90              NOP                         DS:0018  01 01 01 00 02 FF FF FF
010A 90              NOP                         DS:0020  FF FF FF FF FF FF FF FF
010B 49              DEC     CX                  DS:0028  FF FF FF FF EB 19 C0 11
010C 75FB            JNZ     0109                DS:0030  A2 01 14 00 18 00 F5 19
010E 4B              DEC     BX                  DS:0038  FF FF FF FF 00 00 00 00
010F 75F5            JNZ     0106                DS:0040  05 00 00 00 00 00 00 00
0111 48              DEC     AX                  DS:0048  00 00 00 00 00 00 00 00

2          0  1  2  3  4  5  6  7    8  9  A  B  C  D  E  F
DS:0000   CD 20 FF 9F 00 EA F0 FE   AD DE 1B 05 C5 06 00 00   = ƒ.Ω≡■   ↓▌..┼...
DS:0010   18 01 10 01 18 01 92 01   01 01 01 00 02 FF FF FF   ......Æ.  .....
DS:0020   FF FF FF FF FF FF FF FF   FF FF FF FF EB 19 C0 11             δ.└.
DS:0030   A2 01 14 00 18 00 F5 19   FF FF FF FF 00 00 00 00   ó.....J.      ....
DS:0040   05 00 00 00 00 00 00 00   00 00 00 00 00 00 00 00   ........  ........

1 Step  2ProcStep 3Retrieve 4Help ON  5BRK Menu 6        7 up  8 dn  9 le 10 ri
```

mov cx, 4

- Moves the value 4 into the CX register.
- CX now holds the value 4.



```
DOSBox 0.74-3, Cpu speed:    3000 cycles, Frameskip  0, Program:    AFD
AX 0002   SI 0000   CS 19F5   IP 0109      Stack +0 0000  Flags 7200
BX 0003   DI 0000   DS 19F5                      +2 20CD
CX 0004   BP 0000   ES 19F5   HS 19F5            +4 9FFF  OF DF IF SF ZF AF PF CF
DX 0000   SP FFFE   SS 19F5   FS 19F5            +6 EA00   0  0  1  0  0  0  0  0

 CMD >                                           1         0  1  2  3  4  5  6  7
                                                 DS:0000  CD 20 FF 9F 00 EA F0 FE
0106 B90400          MOV     CX,0004             DS:0008  AD DE 1B 05 C5 06 00 00
0109 90              NOP                         DS:0010  18 01 10 01 18 01 92 01
010A 90              NOP                         DS:0018  01 01 01 00 02 FF FF FF
010B 49              DEC     CX                  DS:0020  FF FF FF FF FF FF FF FF
010C 75FB            JNZ     0109                DS:0028  FF FF FF FF EB 19 C0 11
010E 4B              DEC     BX                  DS:0030  A2 01 14 00 18 00 F5 19
010F 75F5            JNZ     0106                DS:0038  FF FF FF FF 00 00 00 00
0111 48              DEC     AX                  DS:0040  05 00 00 00 00 00 00 00
0112 75EF            JNZ     0103                DS:0048  00 00 00 00 00 00 00 00

2          0  1  2  3  4  5  6  7    8  9  A  B  C  D  E  F
DS:0000   CD 20 FF 9F 00 EA F0 FE   AD DE 1B 05 C5 06 00 00   = ƒ.Ω≡■   ↓▌..┼...
DS:0010   18 01 10 01 18 01 92 01   01 01 01 00 02 FF FF FF   ......Æ.  .....
DS:0020   FF FF FF FF FF FF FF FF   FF FF FF FF EB 19 C0 11             δ.└.
DS:0030   A2 01 14 00 18 00 F5 19   FF FF FF FF 00 00 00 00   ó.....J.      ....
DS:0040   05 00 00 00 00 00 00 00   00 00 00 00 00 00 00 00   ........  ........

1 Step  2ProcStep 3Retrieve 4Help ON  5BRK Menu 6        7 up  8 dn  9 le 10 ri
```

Loop Logic

D3 Loop: The program starts the inner-most loop by decrementing cx by one. If cx is not zero, it jumps back to D3 to continue the loop. This process repeats until cx reaches zero.

D2 Loop: Once cx reaches zero, the program decrements bx by one. If bx is not zero, it jumps back to D2 to restart the inner loop with cx initialized to 4 again. This process repeats until bx reaches zero.

D1 Loop: After bx reaches zero, the program decrements ax by one. If ax is not zero, it jumps back to D1 to restart the outer loop with bx initialized to 3 and cx initialized to 4 again. This process repeats until ax reaches zero.

Inner-Most loop (D3):

D3:

dec cx (CX is decremented by 1)

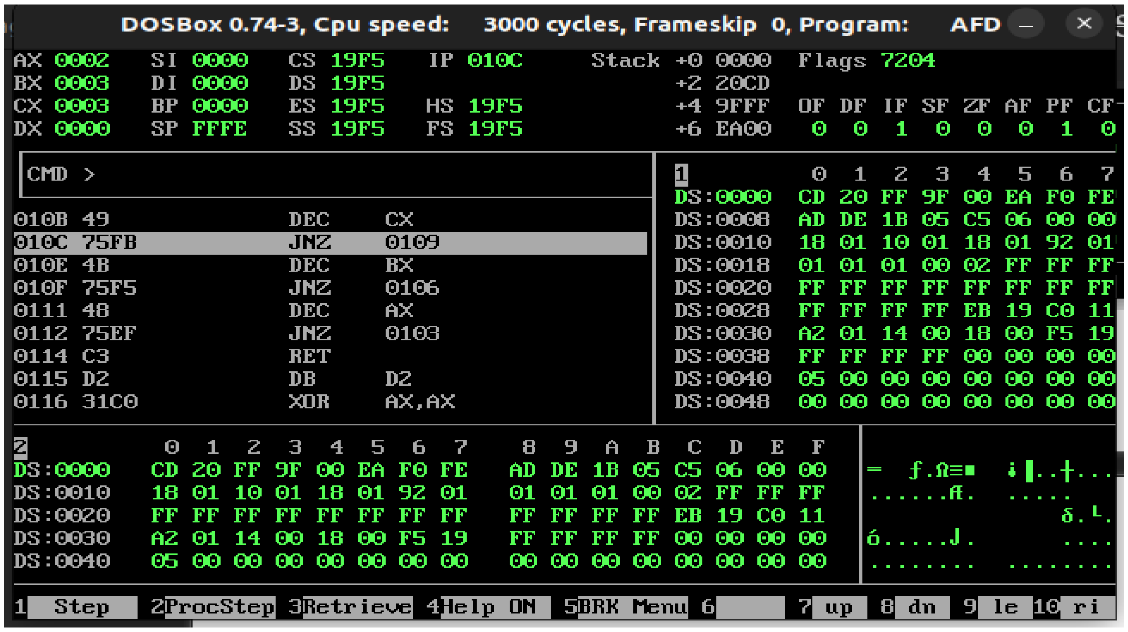
jne D3 (Jump to D3 if CX is NOT equal to zero) [Checks Zero Flag (ZF)]

Inner Loop: This part keeps looping as long as CX is not zero.
Each iteration decrements CX.

First Iteration of inner-most loop:

CX now holds the value 3.

Jumps to D3 because CX is not zero.



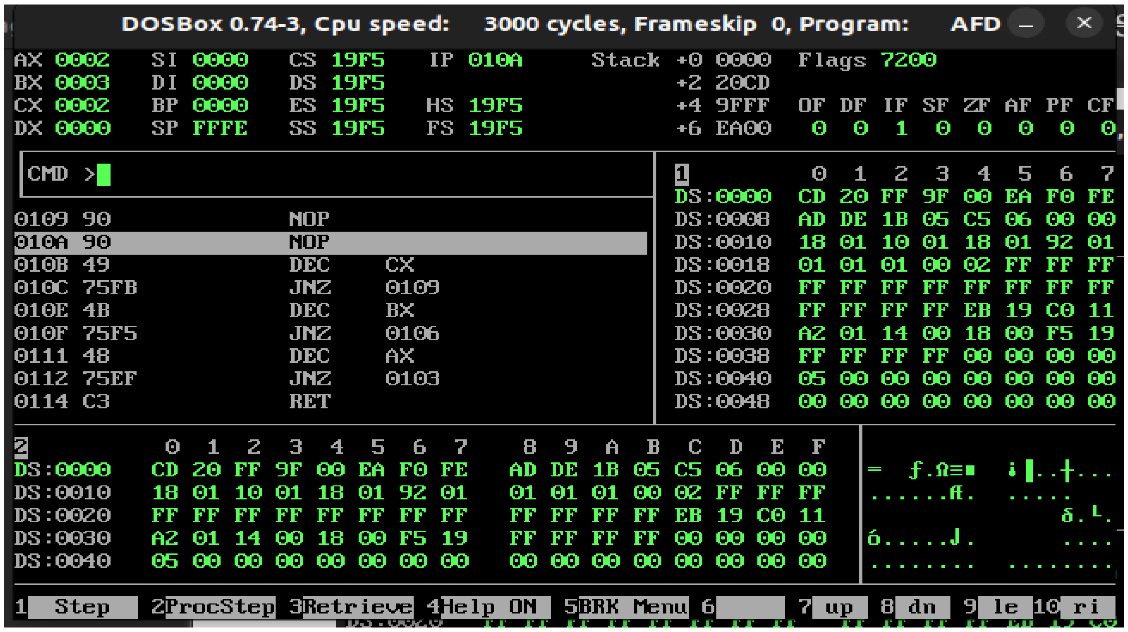
```
DOSBox 0.74-3, Cpu speed:   3000 cycles, Frameskip  0, Program:   AFD
AX 0002   SI 0000   CS 19F5   IP 010C      Stack +0 0000  Flags 7204
BX 0003   DI 0000   DS 19F5                      +2 20CD
CX 0003   BP 0000   ES 19F5   HS 19F5            +4 9FFF  OF DF IF SF ZF AF PF CF
DX 0000   SP FFFE   SS 19F5   FS 19F5            +6 EA00   0  0  1  0  0  0  1  0

CMD >                                           1          0  1  2  3  4  5  6  7
                                                DS:0000   CD 20 FF 9F 00 EA F0 FE
010B 49              DEC     CX                 DS:0008   AD DE 1B 05 C5 06 00 00
010C 75FB            JNZ     0109               DS:0010   18 01 10 01 18 01 92 01
010E 4B              DEC     BX                 DS:0018   01 01 01 00 02 FF FF FF
010F 75F5            JNZ     0106               DS:0020   FF FF FF FF FF FF FF FF
0111 48              DEC     AX                 DS:0028   FF FF FF FF EB 19 C0 11
0112 75EF            JNZ     0103               DS:0030   A2 01 14 00 18 00 F5 19
0114 C3              RET                        DS:0038   FF FF FF FF 00 00 00 00
0115 D2              DB      D2                 DS:0040   05 00 00 00 00 00 00 00
0116 31C0            XOR     AX,AX              DS:0048   00 00 00 00 00 00 00 00

2          0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
DS:0000   CD 20 FF 9F 00 EA F0 FE  AD DE 1B 05 C5 06 00 00   =  ƒ.Ω≡■  ¡▌..┼...
DS:0010   18 01 10 01 18 01 92 01  01 01 01 00 02 FF FF FF   ......Æ.  .....
DS:0020   FF FF FF FF FF FF FF FF  FF FF FF FF EB 19 C0 11                δ.└.
DS:0030   A2 01 14 00 18 00 F5 19  FF FF FF FF 00 00 00 00   ó.....J.      ....
DS:0040   05 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ........  ........

1 Step  2ProcStep  3Retrieve  4Help ON  5BRK Menu  6     7 up  8 dn  9 le  10 ri
```

Second Iteration of inner-most loop:

DEC CX

CX now holds the value 2.

Jumps to D3 because CX is not zero.



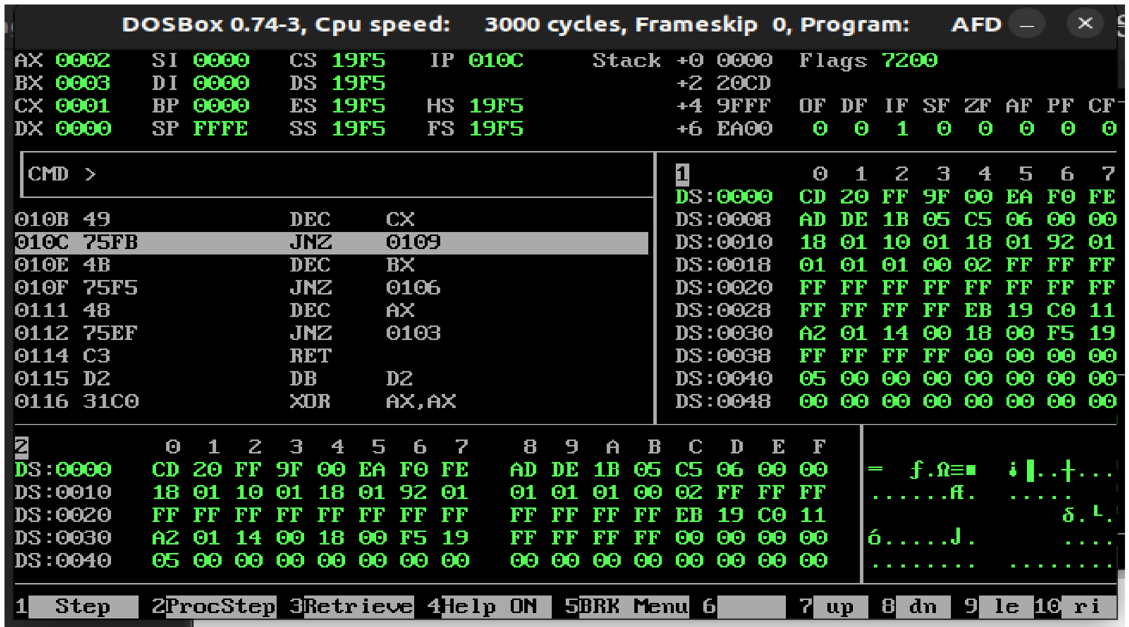
```
DOSBox 0.74-3, Cpu speed:   3000 cycles, Frameskip  0, Program:   AFD
AX 0002   SI 0000   CS 19F5   IP 010A      Stack +0 0000  Flags 7200
BX 0003   DI 0000   DS 19F5                      +2 20CD
CX 0002   BP 0000   ES 19F5   HS 19F5            +4 9FFF  OF DF IF SF ZF AF PF CF
DX 0000   SP FFFE   SS 19F5   FS 19F5            +6 EA00   0  0  1  0  0  0  0  0

 CMD >                                      1        0  1  2  3  4  5  6  7
                                             DS:0000  CD 20 FF 9F 00 EA F0 FE
0109 90                 NOP                  DS:0008  AD DE 1B 05 C5 06 00 00
010A 90                 NOP                  DS:0010  18 01 10 01 18 01 92 01
010B 49                 DEC    CX            DS:0018  01 01 01 00 02 FF FF FF
010C 75FB               JNZ    0109          DS:0020  FF FF FF FF FF FF FF FF
010E 4B                 DEC    BX            DS:0028  FF FF FF FF EB 19 C0 11
010F 75F5               JNZ    0106          DS:0030  A2 01 14 00 18 00 F5 19
0111 48                 DEC    AX            DS:0038  FF FF FF FF 00 00 00 00
0112 75EF               JNZ    0103          DS:0040  05 00 00 00 00 00 00 00
0114 C3                 RET                  DS:0048  00 00 00 00 00 00 00 00

2          0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
DS:0000    CD 20 FF 9F 00 EA F0 FE  AD DE 1B 05 C5 06 00 00
DS:0010    18 01 10 01 18 01 92 01  01 01 01 00 02 FF FF FF
DS:0020    FF FF FF FF FF FF FF FF  FF FF FF FF EB 19 C0 11
DS:0030    A2 01 14 00 18 00 F5 19  FF FF FF FF 00 00 00 00
DS:0040    05 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00

1 Step  2ProcStep 3Retrieve 4Help ON  5BRK Menu 6       7 up  8 dn  9 le 10 ri
```

Third Iteration of inner-most loop:

DEC CX

CX now holds the value 1.

Jumps to D3 because CX is not zero.



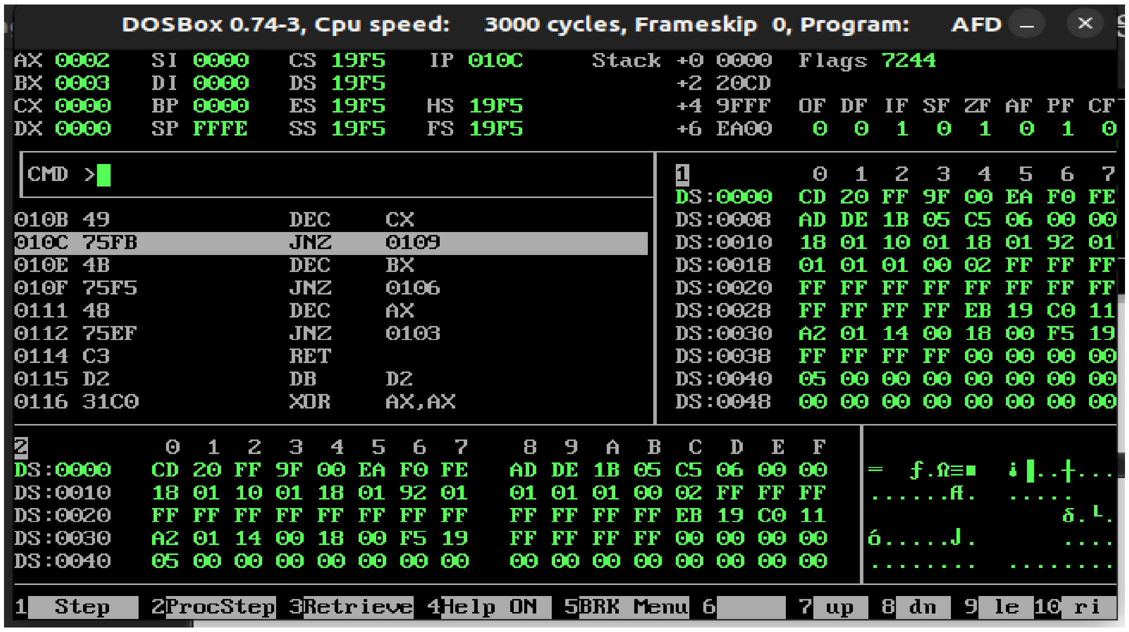
```
DOSBox 0.74-3, Cpu speed:   3000 cycles, Frameskip  0, Program:   AFD
AX 0002   SI 0000   CS 19F5   IP 010C      Stack +0 0000  Flags 7200
BX 0003   DI 0000   DS 19F5                      +2 20CD
CX 0001   BP 0000   ES 19F5   HS 19F5            +4 9FFF  OF DF IF SF ZF AF PF CF
DX 0000   SP FFFE   SS 19F5   FS 19F5            +6 EA00   0  0  1  0  0  0  0  0

 CMD >                                      1        0  1  2  3  4  5  6  7
                                             DS:0000  CD 20 FF 9F 00 EA F0 FE
010B 49                 DEC    CX            DS:0008  AD DE 1B 05 C5 06 00 00
010C 75FB               JNZ    0109          DS:0010  18 01 10 01 18 01 92 01
010E 4B                 DEC    BX            DS:0018  01 01 01 00 02 FF FF FF
010F 75F5               JNZ    0106          DS:0020  FF FF FF FF FF FF FF FF
0111 48                 DEC    AX            DS:0028  FF FF FF FF EB 19 C0 11
0112 75EF               JNZ    0103          DS:0030  A2 01 14 00 18 00 F5 19
0114 C3                 RET                  DS:0038  FF FF FF FF 00 00 00 00
0115 D2                 DB     D2            DS:0040  05 00 00 00 00 00 00 00
0116 31C0               XOR    AX,AX         DS:0048  00 00 00 00 00 00 00 00

2          0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
DS:0000    CD 20 FF 9F 00 EA F0 FE  AD DE 1B 05 C5 06 00 00
DS:0010    18 01 10 01 18 01 92 01  01 01 01 00 02 FF FF FF
DS:0020    FF FF FF FF FF FF FF FF  FF FF FF FF EB 19 C0 11
DS:0030    A2 01 14 00 18 00 F5 19  FF FF FF FF 00 00 00 00
DS:0040    05 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00

1 Step  2ProcStep 3Retrieve 4Help ON  5BRK Menu 6       7 up  8 dn  9 le 10 ri
```

Fourth Iteration of inner-most loop:

DEC CX

CX now holds the value 0.

goes to the next step DEC BX.



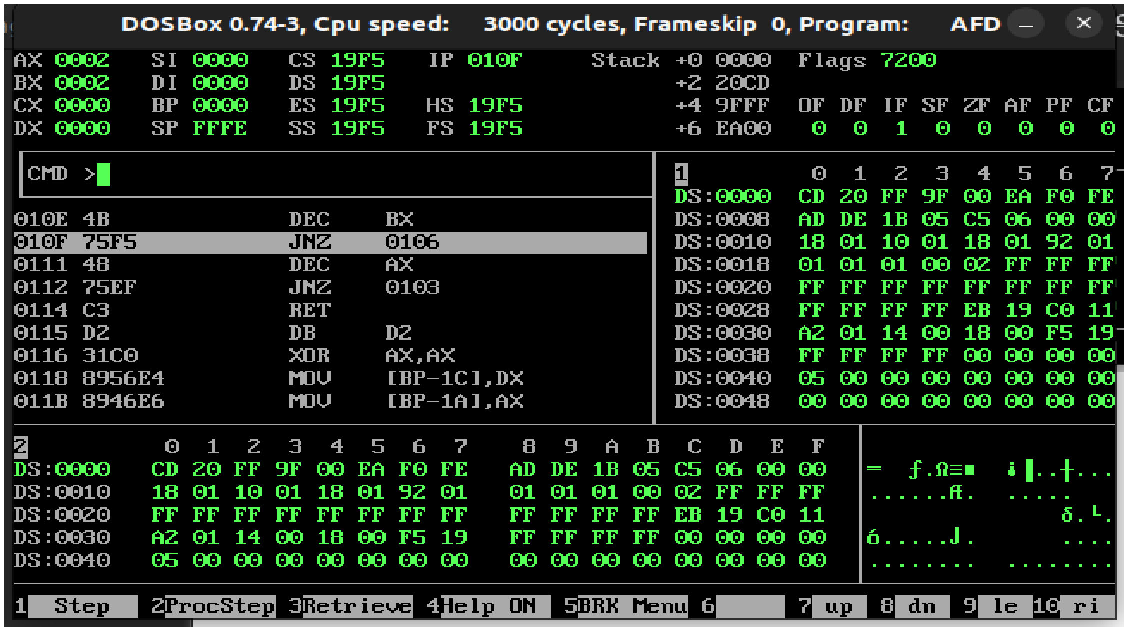
```
DOSBox 0.74-3, Cpu speed:    3000 cycles, Frameskip  0, Program:    AFD
AX 0002   SI 0000   CS 19F5   IP 010C      Stack +0 0000  Flags 7244
BX 0003   DI 0000   DS 19F5                      +2 20CD
CX 0000   BP 0000   ES 19F5   HS 19F5            +4 9FFF  OF DF IF SF ZF AF PF CF
DX 0000   SP FFFE   SS 19F5   FS 19F5            +6 EA00   0  0  1  0  1  0  1  0

 CMD >                                           1          0  1  2  3  4  5  6  7
                                                 DS:0000   CD 20 FF 9F 00 EA F0 FE
010B 49                DEC     CX                DS:0008   AD DE 1B 05 C5 06 00 00
010C 75FB              JNZ     0109              DS:0010   18 01 10 01 18 01 92 01
010E 4B                DEC     BX                DS:0018   01 01 01 00 02 FF FF FF
010F 75F5              JNZ     0106              DS:0020   FF FF FF FF FF FF FF FF
0111 48                DEC     AX                DS:0028   FF FF FF FF EB 19 C0 11
0112 75EF              JNZ     0103              DS:0030   A2 01 14 00 18 00 F5 19
0114 C3                RET                       DS:0038   FF FF FF FF 00 00 00 00
0115 D2                DB      D2                DS:0040   05 00 00 00 00 00 00 00
0116 31C0              XOR     AX,AX             DS:0048   00 00 00 00 00 00 00 00

2         0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
DS:0000  CD 20 FF 9F 00 EA F0 FE  AD DE 1B 05 C5 06 00 00   = ƒ.Ω≡■   ¡▌..┼...
DS:0010  18 01 10 01 18 01 92 01  01 01 01 00 02 FF FF FF   ......Æ.   .....
DS:0020  FF FF FF FF FF FF FF FF  FF FF FF FF EB 19 C0 11                δ.└.
DS:0030  A2 01 14 00 18 00 F5 19  FF FF FF FF 00 00 00 00   ó.....J.       ....
DS:0040  05 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ........   ........

1 Step  2ProcStep 3Retrieve 4Help ON  5BRK Menu 6      7 up  8 dn  9 le 10 ri
```

Inner loop (D2):

First Iteration of Inner loop:
DEC BX (BX is decremented by 1, outside D3)

jne D2 (Jump to D2 if BX is NOT equal to zero) [Checks Zero Flag (ZF)]

After CX reaches zero in D3 (ZF is set), BX is decremented by 1.
BX now holds the value 2.



```
DOSBox 0.74-3, Cpu speed:    3000 cycles, Frameskip  0, Program:    AFD
AX 0002   SI 0000   CS 19F5   IP 010F      Stack +0 0000  Flags 7200
BX 0002   DI 0000   DS 19F5                      +2 20CD
CX 0000   BP 0000   ES 19F5   HS 19F5            +4 9FFF  OF DF IF SF ZF AF PF CF
DX 0000   SP FFFE   SS 19F5   FS 19F5            +6 EA00   0  0  1  0  0  0  0  0

 CMD >                                           1          0  1  2  3  4  5  6  7
                                                 DS:0000   CD 20 FF 9F 00 EA F0 FE
010E 4B                DEC     BX                DS:0008   AD DE 1B 05 C5 06 00 00
010F 75F5              JNZ     0106              DS:0010   18 01 10 01 18 01 92 01
0111 48                DEC     AX                DS:0018   01 01 01 00 02 FF FF FF
0112 75EF              JNZ     0103              DS:0020   FF FF FF FF FF FF FF FF
0114 C3                RET                       DS:0028   FF FF FF FF EB 19 C0 11
0115 D2                DB      D2                DS:0030   A2 01 14 00 18 00 F5 19
0116 31C0              XOR     AX,AX             DS:0038   FF FF FF FF 00 00 00 00
0118 8956E4            MOV     [BP-1C],DX        DS:0040   05 00 00 00 00 00 00 00
011B 8946E6            MOV     [BP-1A],AX        DS:0048   00 00 00 00 00 00 00 00

2         0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
DS:0000  CD 20 FF 9F 00 EA F0 FE  AD DE 1B 05 C5 06 00 00   = ƒ.Ω≡■   ¡▌..┼...
DS:0010  18 01 10 01 18 01 92 01  01 01 01 00 02 FF FF FF   ......Æ.   .....
DS:0020  FF FF FF FF FF FF FF FF  FF FF FF FF EB 19 C0 11                δ.└.
DS:0030  A2 01 14 00 18 00 F5 19  FF FF FF FF 00 00 00 00   ó.....J.       ....
DS:0040  05 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ........   ........

1 Step  2ProcStep 3Retrieve 4Help ON  5BRK Menu 6      7 up  8 dn  9 le 10 ri
```

BX is not zero yet, the program jumps back to **D2**.

Second Iteration of Inner loop:

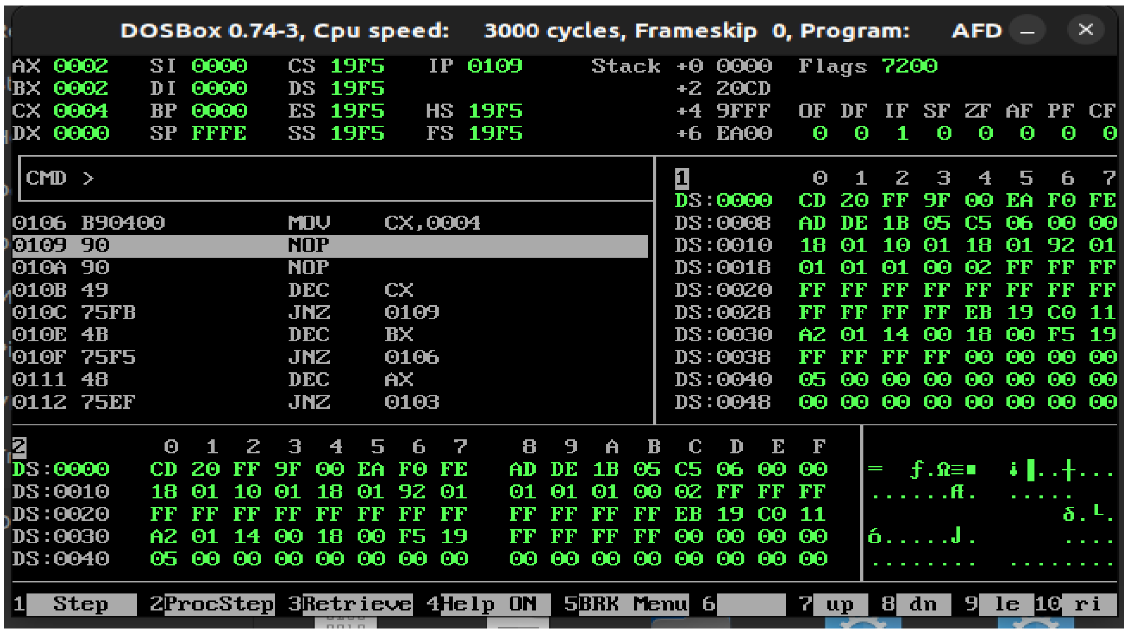
D2:

mov cx, 4 (CX is reloaded with 4, restarting the inner loop)

This resets CX to 4 for the next round of decrements in D3 (inner loop).

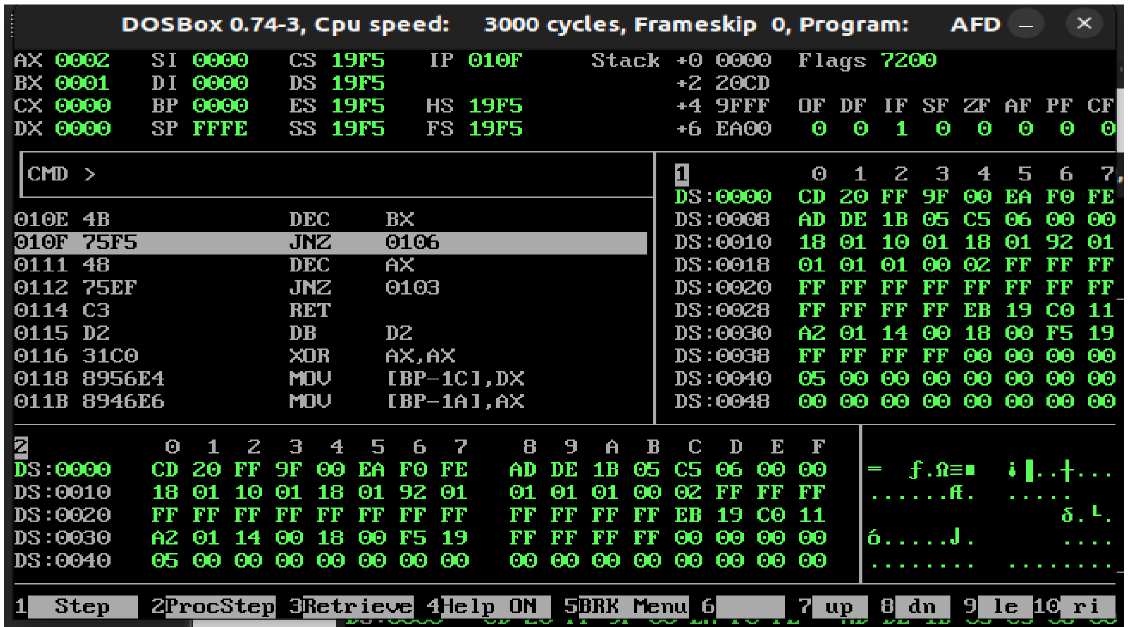
Inner loop keeps looping as long as CX is not zero.

Each iteration decrements CX.



```
DOSBox 0.74-3, Cpu speed:    3000 cycles, Frameskip  0, Program:    AFD
AX 0002   SI 0000   CS 19F5   IP 0109      Stack +0 0000  Flags 7200
BX 0002   DI 0000   DS 19F5                      +2 20CD
CX 0004   BP 0000   ES 19F5   HS 19F5            +4 9FFF  OF DF IF SF ZF AF PF CF
DX 0000   SP FFFE   SS 19F5   FS 19F5            +6 EA00   0  0  1  0  0  0  0  0

 CMD >                                    1        0  1  2  3  4  5  6  7
                                          DS:0000  CD 20 FF 9F 00 EA F0 FE
0106 B90400         MOV    CX,0004        DS:0008  AD DE 1B 05 C5 06 00 00
0109 90             NOP                   DS:0010  18 01 10 01 18 01 92 01
010A 90             NOP                   DS:0018  01 01 01 00 02 FF FF FF
010B 49             DEC    CX             DS:0020  FF FF FF FF FF FF FF FF
010C 75FB           JNZ    0109           DS:0028  FF FF FF FF EB 19 C0 11
010E 4B             DEC    BX             DS:0030  A2 01 14 00 18 00 F5 19
010F 75F5           JNZ    0106           DS:0038  FF FF FF FF 00 00 00 00
0111 48             DEC    AX             DS:0040  05 00 00 00 00 00 00 00
0112 75EF           JNZ    0103           DS:0048  00 00 00 00 00 00 00 00

2         0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
DS:0000   CD 20 FF 9F 00 EA F0 FE  AD DE 1B 05 C5 06 00 00   = ƒ.Ω≡■  ↓▐..┼...
DS:0010   18 01 10 01 18 01 92 01  01 01 01 00 02 FF FF FF   ......Æ.  .....
DS:0020   FF FF FF FF FF FF FF FF  FF FF FF FF EB 19 C0 11             δ.└.
DS:0030   A2 01 14 00 18 00 F5 19  FF FF FF FF 00 00 00 00   ó.....J.      ....
DS:0040   05 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ........  ........

1 Step  2ProcStep 3Retrieve 4Help ON  5BRK Menu 6      7 up  8 dn  9 le 10 ri
```

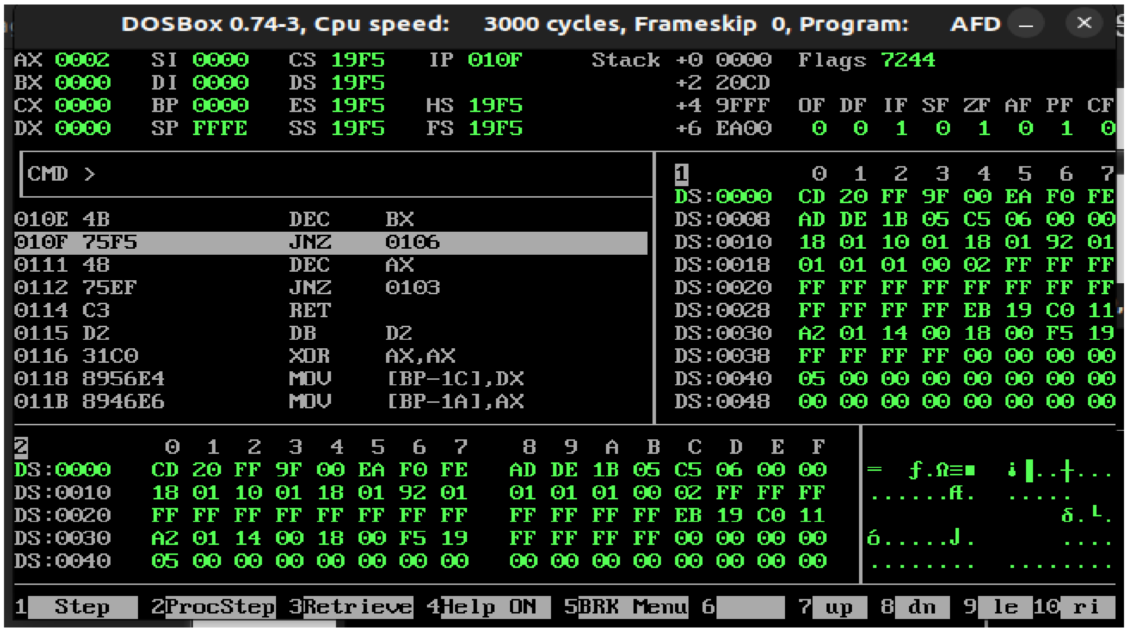
As cx reaches zero, the program decrements bx by one.
Now bx holds the value 1.
BX is not zero yet, the program jumps back to **D2 again restarting the inner loop**.



```
DOSBox 0.74-3, Cpu speed:    3000 cycles, Frameskip  0, Program:    AFD
AX 0002   SI 0000   CS 19F5   IP 010F      Stack +0 0000  Flags 7200
BX 0001   DI 0000   DS 19F5                      +2 20CD
CX 0000   BP 0000   ES 19F5   HS 19F5            +4 9FFF  OF DF IF SF ZF AF PF CF
DX 0000   SP FFFE   SS 19F5   FS 19F5            +6 EA00   0  0  1  0  0  0  0  0

 CMD >                                    1        0  1  2  3  4  5  6  7
                                          DS:0000  CD 20 FF 9F 00 EA F0 FE
010E 4B             DEC    BX             DS:0008  AD DE 1B 05 C5 06 00 00
010F 75F5           JNZ    0106           DS:0010  18 01 10 01 18 01 92 01
0111 48             DEC    AX             DS:0018  01 01 01 00 02 FF FF FF
0112 75EF           JNZ    0103           DS:0020  FF FF FF FF FF FF FF FF
0114 C3             RET                   DS:0028  FF FF FF FF EB 19 C0 11
0115 D2             DB     D2             DS:0030  A2 01 14 00 18 00 F5 19
0116 31C0           XOR    AX,AX          DS:0038  FF FF FF FF 00 00 00 00
0118 8956E4         MOV    [BP-1C],DX     DS:0040  05 00 00 00 00 00 00 00
011B 8946E6         MOV    [BP-1A],AX     DS:0048  00 00 00 00 00 00 00 00

2         0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
DS:0000   CD 20 FF 9F 00 EA F0 FE  AD DE 1B 05 C5 06 00 00   = ƒ.Ω≡■  ↓▐..┼...
DS:0010   18 01 10 01 18 01 92 01  01 01 01 00 02 FF FF FF   ......Æ.  .....
DS:0020   FF FF FF FF FF FF FF FF  FF FF FF FF EB 19 C0 11             δ.└.
DS:0030   A2 01 14 00 18 00 F5 19  FF FF FF FF 00 00 00 00   ó.....J.      ....
DS:0040   05 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ........  ........

1 Step  2ProcStep 3Retrieve 4Help ON  5BRK Menu 6      7 up  8 dn  9 le 10 ri
```

Third Iteration of Inner loop:
As cx reaches zero, the program decrements bx by one.

Now bx holds the value 0.

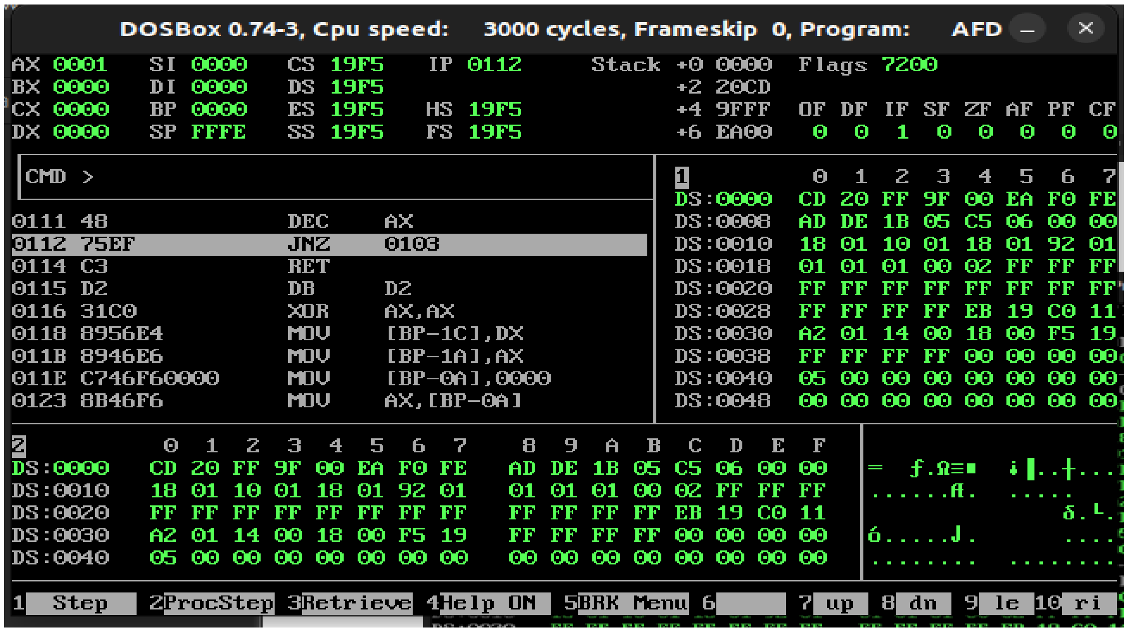


```
DOSBox 0.74-3, Cpu speed:    3000 cycles, Frameskip  0, Program:    AFD
AX 0002   SI 0000   CS 19F5   IP 010F     Stack +0 0000  Flags 7244
BX 0000   DI 0000   DS 19F5                     +2 20CD
CX 0000   BP 0000   ES 19F5   HS 19F5           +4 9FFF   OF DF IF SF ZF AF PF CF
DX 0000   SP FFFE   SS 19F5   FS 19F5           +6 EA00    0  0  1  0  1  0  1  0
 CMD >                                        1          0  1  2  3  4  5  6  7
                                              DS:0000   CD 20 FF 9F 00 EA F0 FE
010E 4B                 DEC     BX            DS:0008   AD DE 1B 05 C5 06 00 00
010F 75F5               JNZ     0106          DS:0010   18 01 10 01 18 01 92 01
0111 48                 DEC     AX            DS:0018   01 01 01 00 02 FF FF FF
0112 75EF               JNZ     0103          DS:0020   FF FF FF FF FF FF FF FF
0114 C3                 RET                   DS:0028   FF FF FF FF EB 19 C0 11
0115 D2                 DB      D2            DS:0030   A2 01 14 00 18 00 F5 19
0116 31C0               XOR     AX,AX         DS:0038   FF FF FF FF 00 00 00 00
0118 8956E4             MOV     [BP-1C],DX    DS:0040   05 00 00 00 00 00 00 00
011B 8946E6             MOV     [BP-1A],AX    DS:0048   00 00 00 00 00 00 00 00
2          0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
DS:0000   CD 20 FF 9F 00 EA F0 FE  AD DE 1B 05 C5 06 00 00  = ƒ.Ω≡■  ↓▌..┼...
DS:0010   18 01 10 01 18 01 92 01  01 01 01 00 02 FF FF FF  ......Æ.  .....
DS:0020   FF FF FF FF FF FF FF FF  FF FF FF FF EB 19 C0 11              δ.└.
DS:0030   A2 01 14 00 18 00 F5 19  FF FF FF FF 00 00 00 00  ó.....J.      ....
DS:0040   05 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ........  ........
1 Step  2ProcStep 3Retrieve 4Help ON  5BRK Menu 6      7 up  8 dn  9 le 10 ri
```

Outer Most Loop(D1):
jne D1

(**Jump to D1 if AX is NOT equal to zero**) **[Checks Zero Flag (ZF)]**After the inner loop finishes (BX becomes zero), AX is decremented by 1.

As AX is not zero yet, the program jumps back to **D1.**



```
DOSBox 0.74-3, Cpu speed:    3000 cycles, Frameskip  0, Program:    AFD
AX 0001   SI 0000   CS 19F5   IP 0112     Stack +0 0000  Flags 7200
BX 0000   DI 0000   DS 19F5                     +2 20CD
CX 0000   BP 0000   ES 19F5   HS 19F5           +4 9FFF   OF DF IF SF ZF AF PF CF
DX 0000   SP FFFE   SS 19F5   FS 19F5           +6 EA00    0  0  1  0  0  0  0  0
 CMD >                                        1          0  1  2  3  4  5  6  7
                                              DS:0000   CD 20 FF 9F 00 EA F0 FE
0111 48                 DEC     AX            DS:0008   AD DE 1B 05 C5 06 00 00
0112 75EF               JNZ     0103          DS:0010   18 01 10 01 18 01 92 01
0114 C3                 RET                   DS:0018   01 01 01 00 02 FF FF FF
0115 D2                 DB      D2            DS:0020   FF FF FF FF FF FF FF FF
0116 31C0               XOR     AX,AX         DS:0028   FF FF FF FF EB 19 C0 11
0118 8956E4             MOV     [BP-1C],DX    DS:0030   A2 01 14 00 18 00 F5 19
011B 8946E6             MOV     [BP-1A],AX    DS:0038   FF FF FF FF 00 00 00 00
011E C746F60000         MOV     [BP-0A],0000  DS:0040   05 00 00 00 00 00 00 00
0123 8B46F6             MOV     AX,[BP-0A]    DS:0048   00 00 00 00 00 00 00 00
2          0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
DS:0000   CD 20 FF 9F 00 EA F0 FE  AD DE 1B 05 C5 06 00 00  = ƒ.Ω≡■  ↓▌..┼...
DS:0010   18 01 10 01 18 01 92 01  01 01 01 00 02 FF FF FF  ......Æ.  .....
DS:0020   FF FF FF FF FF FF FF FF  FF FF FF FF EB 19 C0 11              δ.└.
DS:0030   A2 01 14 00 18 00 F5 19  FF FF FF FF 00 00 00 00  ó.....J.      ....
DS:0040   05 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ........  ........
1 Step  2ProcStep 3Retrieve 4Help ON  5BRK Menu 6      7 up  8 dn  9 le 10 ri
```

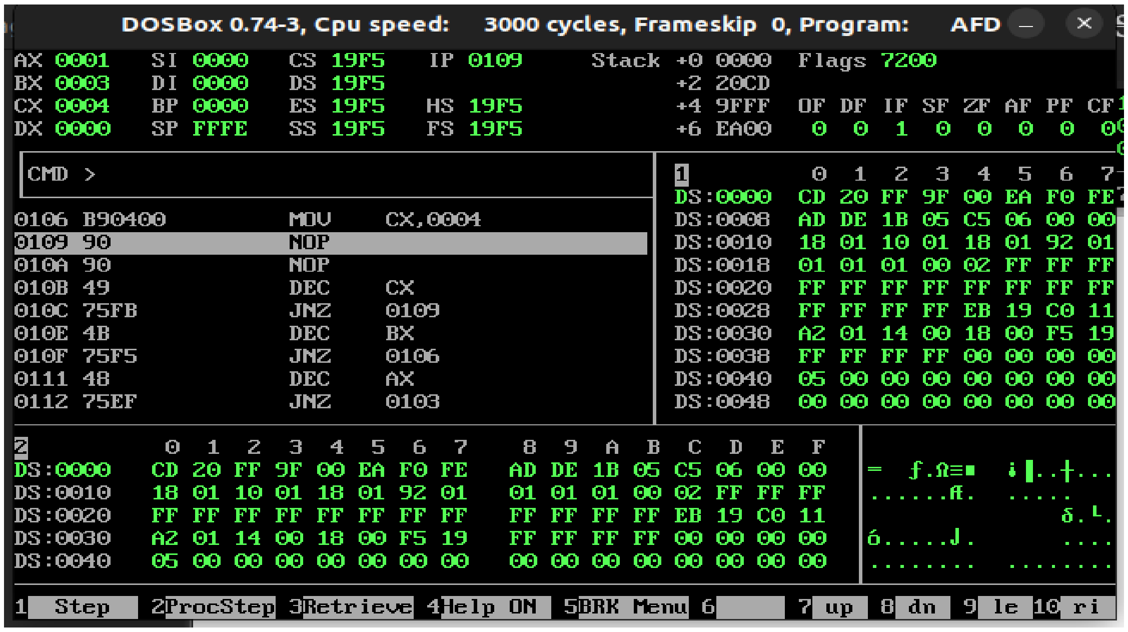
D1:

First Iteration of Outer loop:

mov bx, 3 (BX is reloaded with 3, restarting the loop of D2)

This resets BX to 3 for the next outer loop iteration as AX is not zero.

And again the same steps repeated.

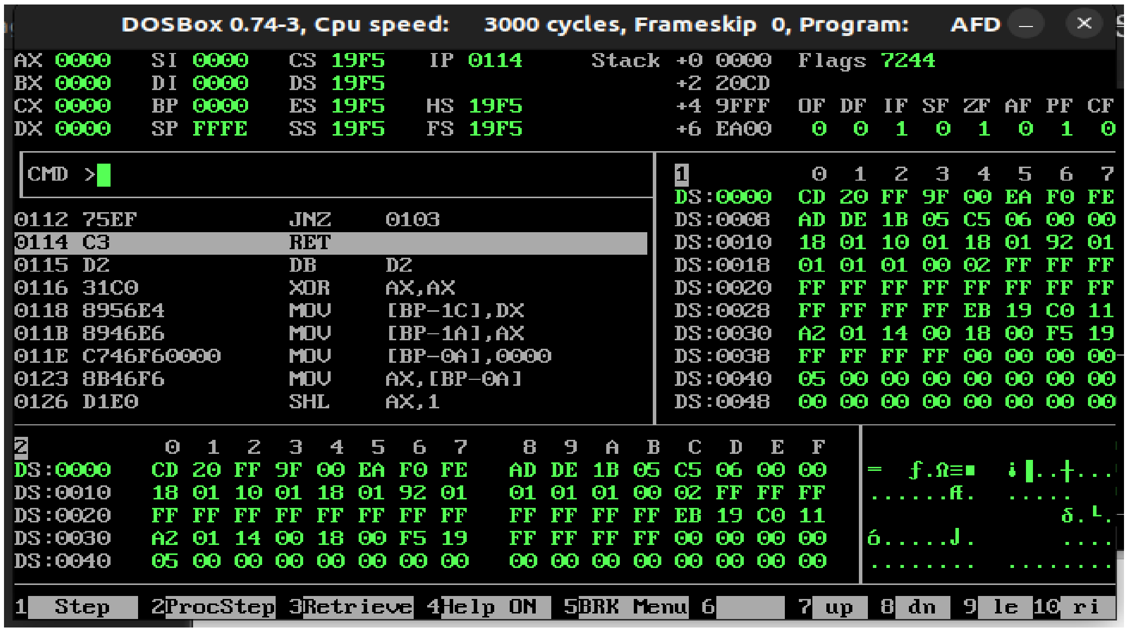


```
DOSBox 0.74-3, Cpu speed:    3000 cycles, Frameskip  0, Program:    AFD
AX 0001   SI 0000   CS 19F5   IP 0109      Stack +0 0000  Flags 7200
BX 0003   DI 0000   DS 19F5                      +2 20CD
CX 0004   BP 0000   ES 19F5   HS 19F5            +4 9FFF  OF DF IF SF ZF AF PF CF
DX 0000   SP FFFE   SS 19F5   FS 19F5            +6 EA00   0  0  1  0  0  0  0  0

 CMD >                                          1          0  1  2  3  4  5  6  7
                                                DS:0000   CD 20 FF 9F 00 EA F0 FE
0106 B90400          MOV     CX,0004            DS:0008   AD DE 1B 05 C5 06 00 00
0109 90              NOP                        DS:0010   18 01 10 01 18 01 92 01
010A 90              NOP                        DS:0018   01 01 01 00 02 FF FF FF
010B 49              DEC     CX                 DS:0020   FF FF FF FF FF FF FF FF
010C 75FB            JNZ     0109               DS:0028   FF FF FF FF EB 19 C0 11
010E 4B              DEC     BX                 DS:0030   A2 01 14 00 18 00 F5 19
010F 75F5            JNZ     0106               DS:0038   FF FF FF FF 00 00 00 00
0111 48              DEC     AX                 DS:0040   05 00 00 00 00 00 00 00
0112 75EF            JNZ     0103               DS:0048   00 00 00 00 00 00 00 00

2         0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
DS:0000   CD 20 FF 9F 00 EA F0 FE  AD DE 1B 05 C5 06 00 00   = ƒ.Ω≡■  ¡▌..┼...
DS:0010   18 01 10 01 18 01 92 01  01 01 01 00 02 FF FF FF   ......Æ.  .....
DS:0020   FF FF FF FF FF FF FF FF  FF FF FF FF EB 19 C0 11                δ.└.
DS:0030   A2 01 14 00 18 00 F5 19  FF FF FF FF 00 00 00 00   ó.....J.      ....
DS:0040   05 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ........  ........

1 Step  2ProcStep 3Retrieve 4Help ON  5BRK Menu 6       7 up  8 dn  9 le 10 ri
```

Second Iteration of Outer loop:

When BX becomes zero it again decrements AX by 1.

Now AX become zero (ZF is set after the decrement in the outer loop), the jne instructions fail to jump, and the program reaches the end.



```
DOSBox 0.74-3, Cpu speed:    3000 cycles, Frameskip  0, Program:    AFD
AX 0000   SI 0000   CS 19F5   IP 0114      Stack +0 0000  Flags 7244
BX 0000   DI 0000   DS 19F5                      +2 20CD
CX 0000   BP 0000   ES 19F5   HS 19F5            +4 9FFF  OF DF IF SF ZF AF PF CF
DX 0000   SP FFFE   SS 19F5   FS 19F5            +6 EA00   0  0  1  0  1  0  1  0

 CMD >                                          1          0  1  2  3  4  5  6  7
                                                DS:0000   CD 20 FF 9F 00 EA F0 FE
0112 75EF            JNZ     0103               DS:0008   AD DE 1B 05 C5 06 00 00
0114 C3              RET                        DS:0010   18 01 10 01 18 01 92 01
0115 D2              DB      D2                 DS:0018   01 01 01 00 02 FF FF FF
0116 31C0            XOR     AX,AX              DS:0020   FF FF FF FF FF FF FF FF
0118 8956E4          MOV     [BP-1C],DX         DS:0028   FF FF FF FF EB 19 C0 11
011B 8946E6          MOV     [BP-1A],AX         DS:0030   A2 01 14 00 18 00 F5 19
011E C746F60000      MOV     [BP-0A],0000       DS:0038   FF FF FF FF 00 00 00 00
0123 8B46F6          MOV     AX,[BP-0A]         DS:0040   05 00 00 00 00 00 00 00
0126 D1E0            SHL     AX,1               DS:0048   00 00 00 00 00 00 00 00

2         0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
DS:0000   CD 20 FF 9F 00 EA F0 FE  AD DE 1B 05 C5 06 00 00   = ƒ.Ω≡■  ¡▌..┼...
DS:0010   18 01 10 01 18 01 92 01  01 01 01 00 02 FF FF FF   ......Æ.  .....
DS:0020   FF FF FF FF FF FF FF FF  FF FF FF FF EB 19 C0 11                δ.└.
DS:0030   A2 01 14 00 18 00 F5 19  FF FF FF FF 00 00 00 00   ó.....J.      ....
DS:0040   05 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ........  ........

1 Step  2ProcStep 3Retrieve 4Help ON  5BRK Menu 6       7 up  8 dn  9 le 10 ri
```