

```

52 ; -----
53 ; A simple game using CMP
54 ; -----
55 game:
56     mov dx, offset msg1 ; Load message address into DX
57     mov ah, 9           ; DOS interrupt for string output
58     int 21h             ; Print the message
59
60     mov ah, 1           ; DOS interrupt to read a character into AL
61     int 21h             ; Input character
62
63     cmp al, '0'         ; Check if input is less than '0'
64     jb stop             ; If below '0', exit the game
65
66     cmp al, '9'         ; Check if input is greater than '9'
67     ja stop             ; If above '9', exit the game
68
69     cmp al, '5'         ; Compare input with '5'
70     jb below            ; If less than '5', jump to below
71     ja above            ; If greater than '5', jump to above
72     mov dx, offset equal_5 ; If equal to '5', load corresponding message
73     jmp print           ; Jump to print the message
74
75 below:
76     mov dx, offset below_5 ; Load message for below 5
77     jmp print             ; Jump to print the message
78
79 above:
80     mov dx, offset above_5 ; Load message for above 5
81
82 print:
83     mov ah, 9           ; DOS interrupt for string output
84     int 21h             ; Print the message
85     jmp game            ; Loop back to the game
86
87 stop:
88     ret                 ; Return and exit the program
89
90 ; -----
91 ; Data Section
92 ; -----
93 msg1 db "Enter a number (0-9) or any other character to exit: $"
94 equal_5 db "The number is five! (equal)", 0Dh, 0Ah, "$"
95 below_5 db "The number is below five!", 0Dh, 0Ah, "$"
96 above_5 db "The number is above five!", 0Dh, 0Ah, "$"
97

```

emu8086 - assembler and microprocessor emulator 4.08

file edit bookmarks assembler emulator math ascii codes help

new open examples save compile emulate calculator convertor options

01 name "flags"
02
03 org 100h
04

emulator: flags.com_

file math debug view external virtual devices virtual drive help

Load reload step back waiting for input stop step delay ms: 0

registers

	H	L
AX	01	24
BX	00	00
CX	00	EA
DX	01	5E
CS	F400	
IP	0200	
SS	0700	
SP	FFF8	
BP	0000	
SI	0000	
DI	0000	
DS	0700	
ES	0700	

F400:0200 F400:0200

emulator screen (80x25 chars)

```
F42 Enter a number <0-9> or any other character to exit: 5The number is five! <equal
F42 >
F42 Enter a number <0-9> or any other character to exit: 4The number is below five!
F42 >
F42 Enter a number <0-9> or any other character to exit: 5The number is five! <equal
F42 >
F42 Enter a number <0-9> or any other character to exit: 4The number is below five!
F42 >
F42 Enter a number <0-9> or any other character to exit: 2The number is below five!
F42 >
F42 Enter a number <0-9> or any other character to exit:
F42
F42
F42
F42
F42
F42
F42
F42
F42
F42
F42
F42
```

clear screen change font 0/16

56 mov dx, offset msg1 ; L
57 mov ah, 9 ; D
58 int 21h ; P
59
60 mov ah, 1 ; D
61 int 21h ; I
62
63 cmp al, '0' ; C
64 jb stop ; I
65
66 cmp al, '9' ; C
67 ja stop ; I
68
69 cmp al, '5' ; C
70 jb below ; I
71 ja above ; I
72 mov dx, offset equal_5 ; J
73 jmp print ; J

41 mov ah, -
42 mov al, -
43 cmp ah, a
44 nop
45
46 ; 255 is above 1 <unsigned comparison>
47 mov ah, 255 ; Load 255 <-1 in unsigned> into AH
48 mov al, 1 ; Load 1 into AL
49 cmp ah, al ; Compare AH and AL <unsigned>, CF=0
50 nop ; No operation