Program 1:

emu8086 - assembler and microprocessor emulator 4.08

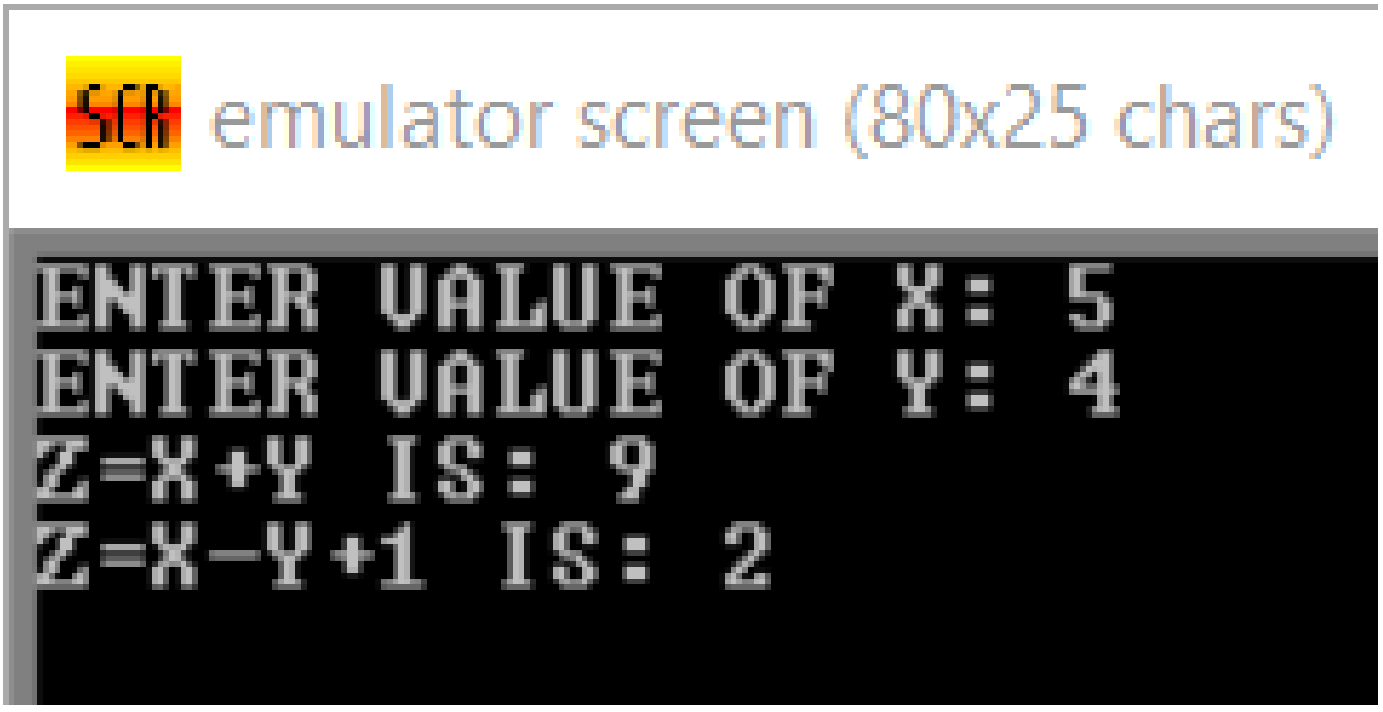
file edit bookmarks assembler emulator math ascii codes help

new open examples save compile emulate calculator convertor options help about

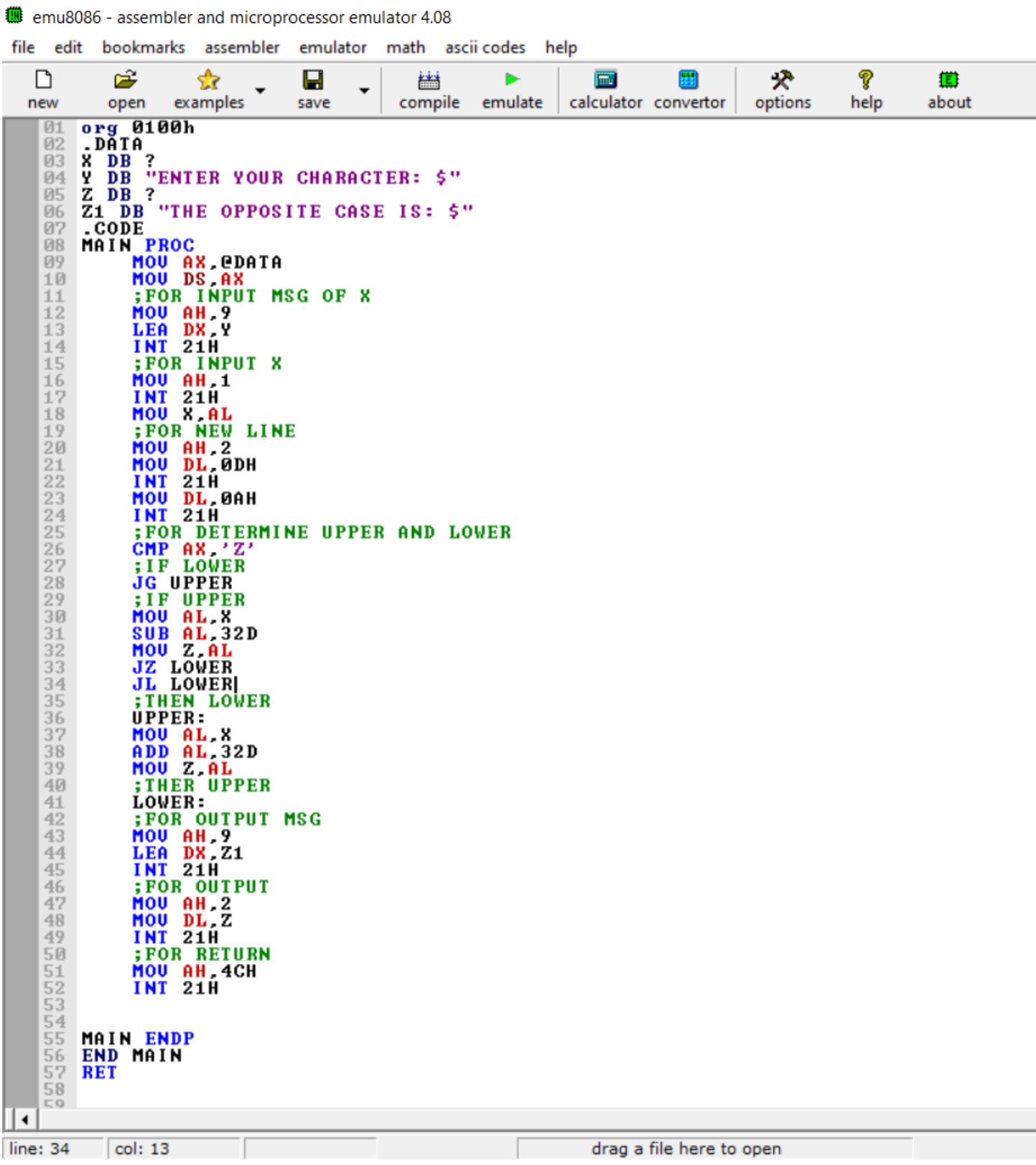
```
01 org 0100h
02 .DATA
03 X DB ?
04 X1 DB "ENTER VALUE OF X: $"
05 Y DB ?
06 Y1 DB "ENTER VALUE OF Y: $"
07 Z1 DB ?
08 Z11 DB "Z=X+Y IS: $"
09 Z2 DB ?
10 Z22 DB "Z=X-Y+1 IS: $"
11 .CODE
12 MAIN PROC
13     MOV AX,@DATA
14     MOV DS,AX
15     ;FOR INPUT MSG OF X
16     MOV AH,9
17     LEA DX,X1
18     INT 21H
19     ;FOR INPUT X
20     MOV AH,1
21     INT 21H
22     MOV X,AL
23     ;FOR NEW LINE
24     MOV AH,2
25     MOV DL,0DH
26     INT 21H
27     MOV DL,0AH
28     INT 21H
29     ;FOR INPUT MSG OF Y
30     MOV AH,9
31     LEA DX,Y1
32     INT 21H
33     ;FOR INPUT Y
34     MOV AH,1
35     INT 21H
36     MOV Y,AL
37     ;FOR NEW LINE
38     MOV AH,2
39     MOV DL,0DH
40     INT 21H
41     MOV DL,0AH
42     INT 21H
43     ;FOR STORE Z=X+Y
44     MOV AL,X
45     ADD AL,Y
46     SUB AL,'0'
47     MOV Z1,AL
48     ;FOR OUTPUT MSG OF Z=X+Y
49     MOV AH,9
50     LEA DX,Z11
51     INT 21H
52     ;FOR OUTPUT Z=X+Y
53     MOV AH,2
54     MOV DL,Z1
55     INT 21H
56     ;FOR NEW LINE
```

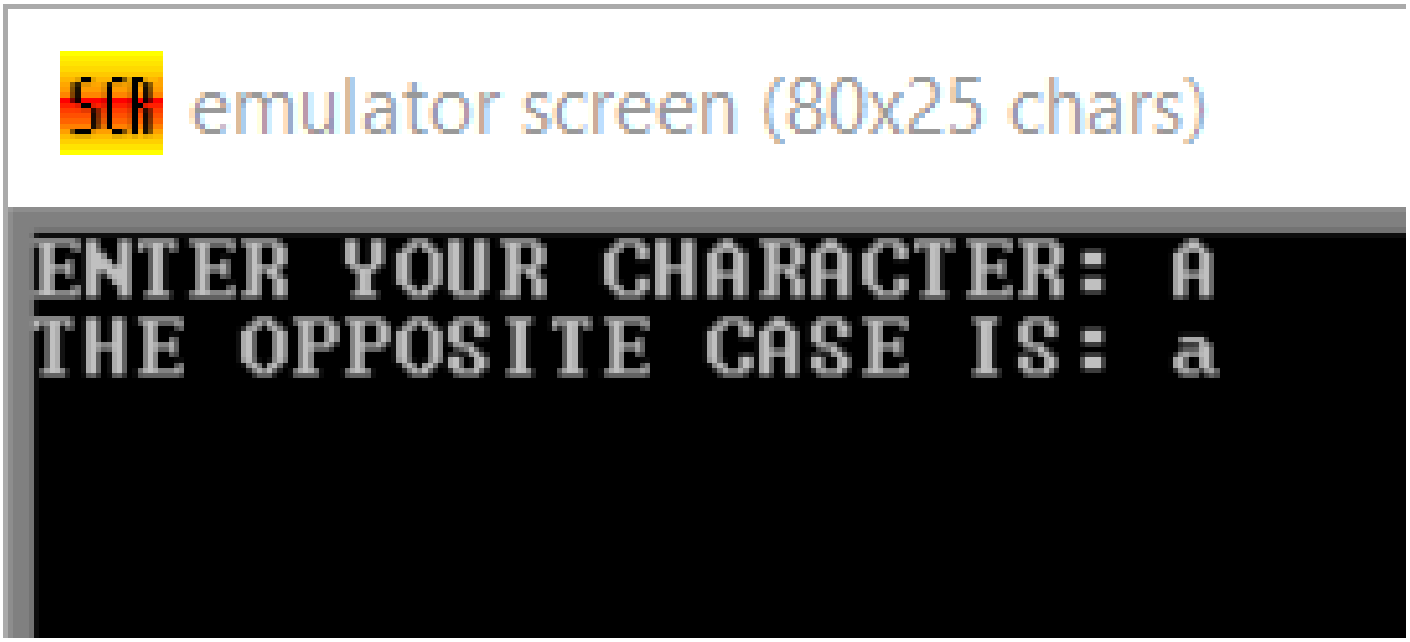
```
56      ;FOR NEW LINE
57      MOV AH,2
58      MOV DL,0DH
59      INT 21H
60      MOV DL,0AH
61      INT 21H
62      ;FOR STORE Z=X-Y+1
63      MOV AL,X
64      SUB AL,Y
65      ADD AL,'0'
66      ADD AL,1D
67      MOV Z2,AL
68      ;FOR OUTPUT MSF OF Z=X-Y+1
69      MOV AH,9
70      LEA DX,Z22
71      INT 21H
72      ;FOR OUTPUT Z=X-Y+1
73      MOV AH,2
74      MOV DL,Z2
75      INT 21H
76      ;FOR RETURN
77      MOV AH,4CH
78      INT 21H
79
80
81      MAIN ENDP
82  END MAIN
83  RET
84
```

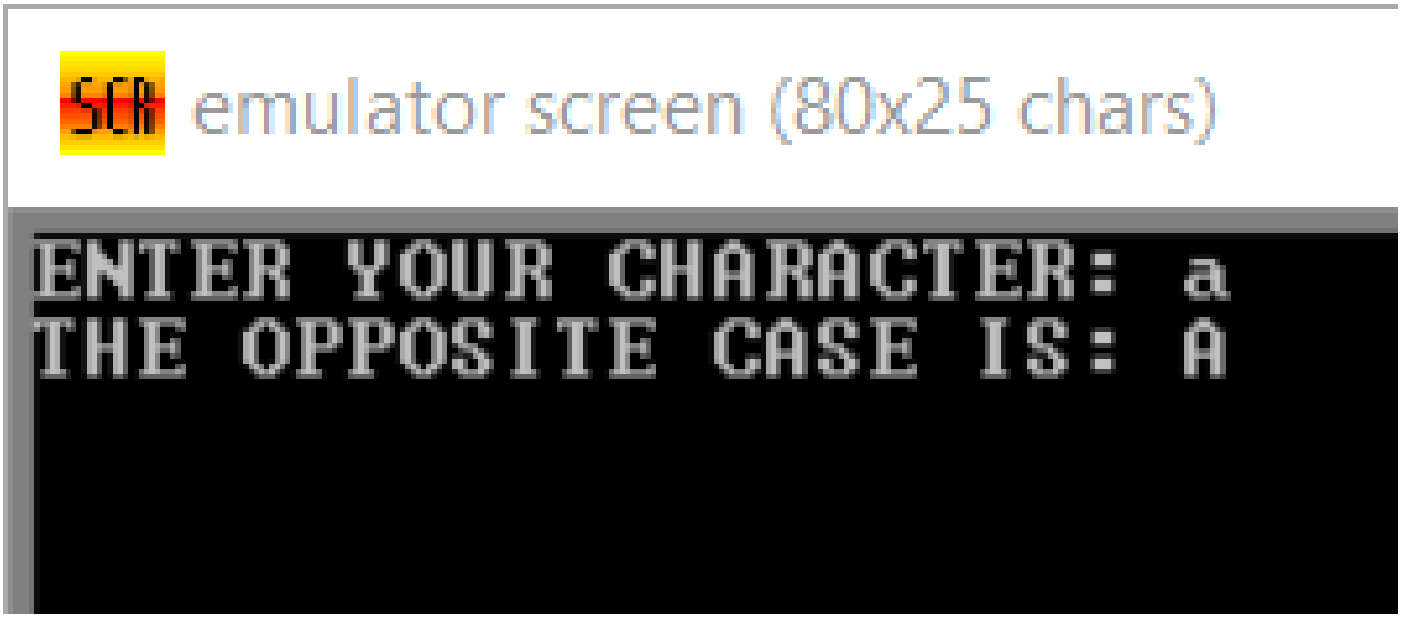
line: 84 | col: 1 | drag a file here to open



Program 2:


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 help about
org 0100h
.DATA
X DB ?
Y DB "ENTER YOUR CHARACTER: $"
Z DB ?
Z1 DB "THE OPPOSITE CASE IS: $"
.CODE
MAIN PROC
MOV AX,@DATA
MOV DS,AX
;FOR INPUT MSG OF X
MOV AH,9
LEA DX,Y
INT 21H
;FOR INPUT X
MOV AH,1
INT 21H
MOV X,AL
;FOR NEW LINE
MOV AH,2
MOV DL,0DH
INT 21H
MOV DL,0AH
INT 21H
;FOR DETERMINE UPPER AND LOWER
CMP AX,'Z'
;IF LOWER
JG UPPER
;IF UPPER
MOV AL,X
SUB AL,32D
MOV Z,AL
JZ LOWER
JL LOWER
;THEN LOWER
UPPER:
MOV AL,X
ADD AL,32D
MOV Z,AL
;THER UPPER
LOWER:
;FOR OUTPUT MSG
MOV AH,9
LEA DX,Z1
INT 21H
;FOR OUTPUT
MOV AH,2
MOV DL,Z
INT 21H
;FOR RETURN
MOV AH,4CH
INT 21H
MAIN ENDP
END MAIN
RET
line: 34 col: 13
drag a file here to open


SCR emulator screen (80x25 chars)
ENTER YOUR CHARACTER: A
THE OPPOSITE CASE IS: a


SCR emulator screen (80x25 chars)
ENTER YOUR CHARACTER: a
THE OPPOSITE CASE IS: A