Alexandria University,

Faculty of Engineering,

Computer and Systems Engineering Dept.

CS222: System programming

Lab (2) Report

Name:

Abdel-Fatah Mohamed Abdel-Fatah. (24)

Content:

1) Simulator.
2) PROGRAMS.

   a. Write a program that searches for a certain byte in a string, if found, put the address of the byte in register A else, put 0xFFFFF.

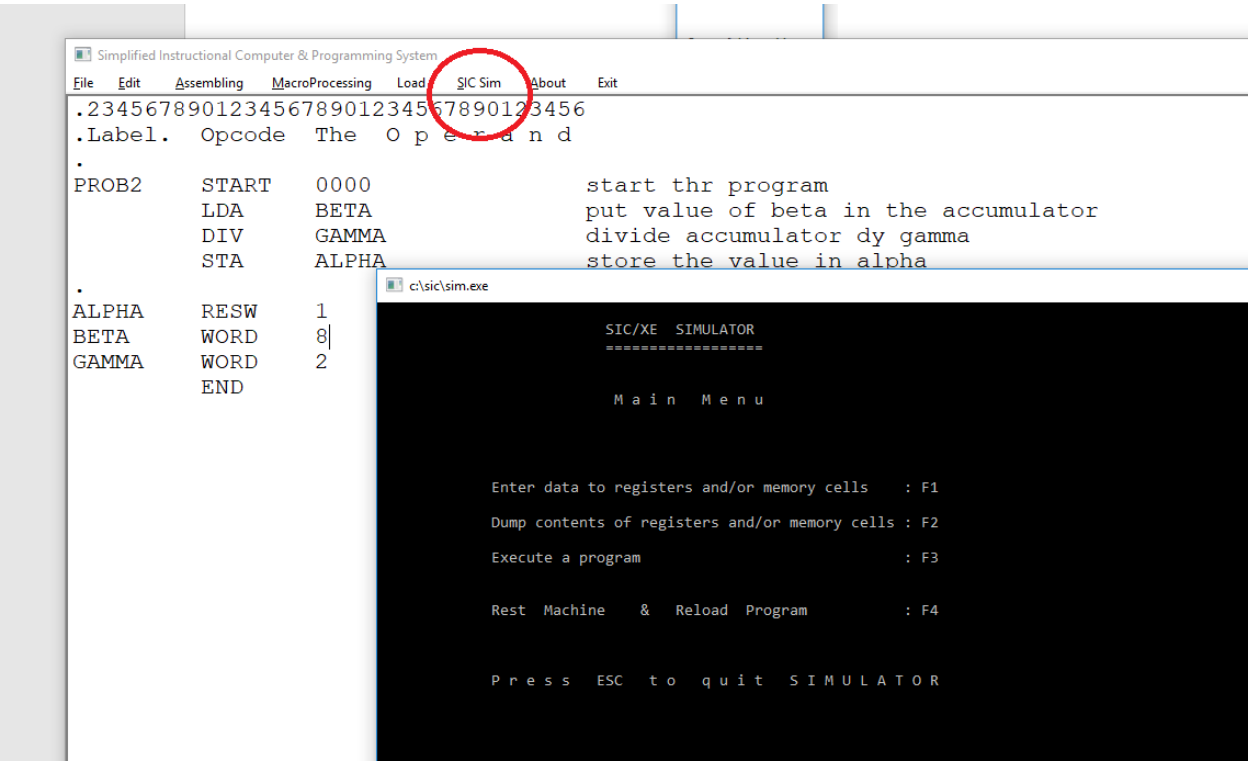
   b. Read a string from an input device and print it reversed to an output device.

   c. Read a string from an input device and print the string to an output device after converting it to UPPER case.

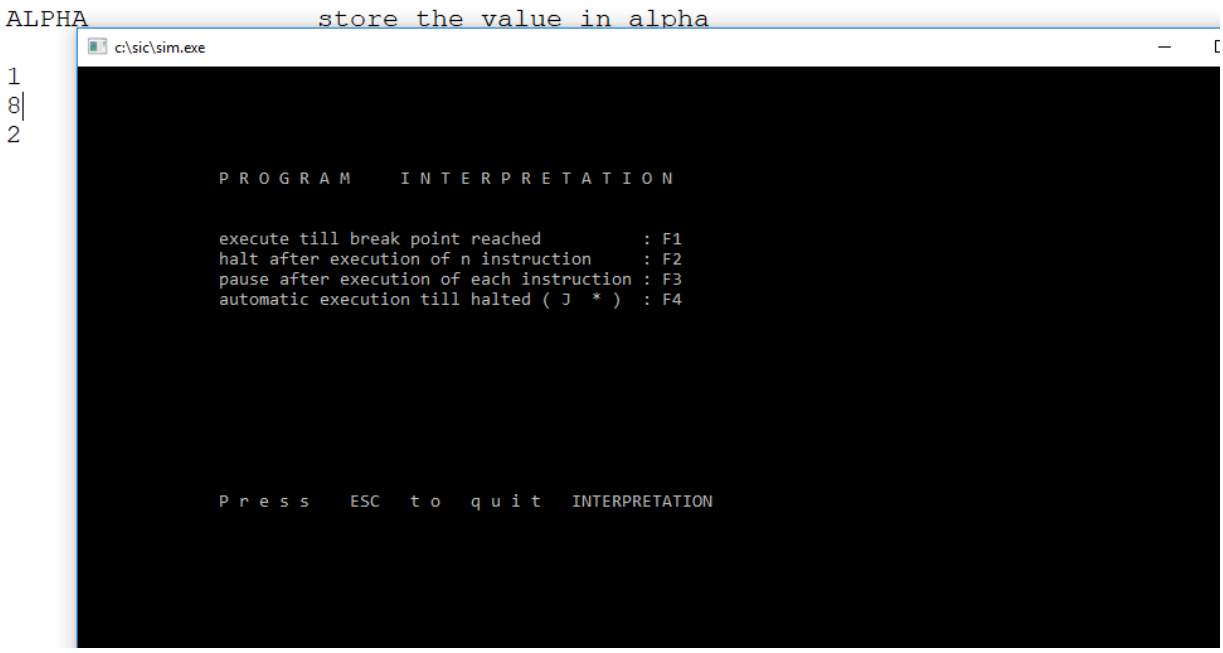
   d. Implement the bubble sort to sort characters in a string.

***Simulator:

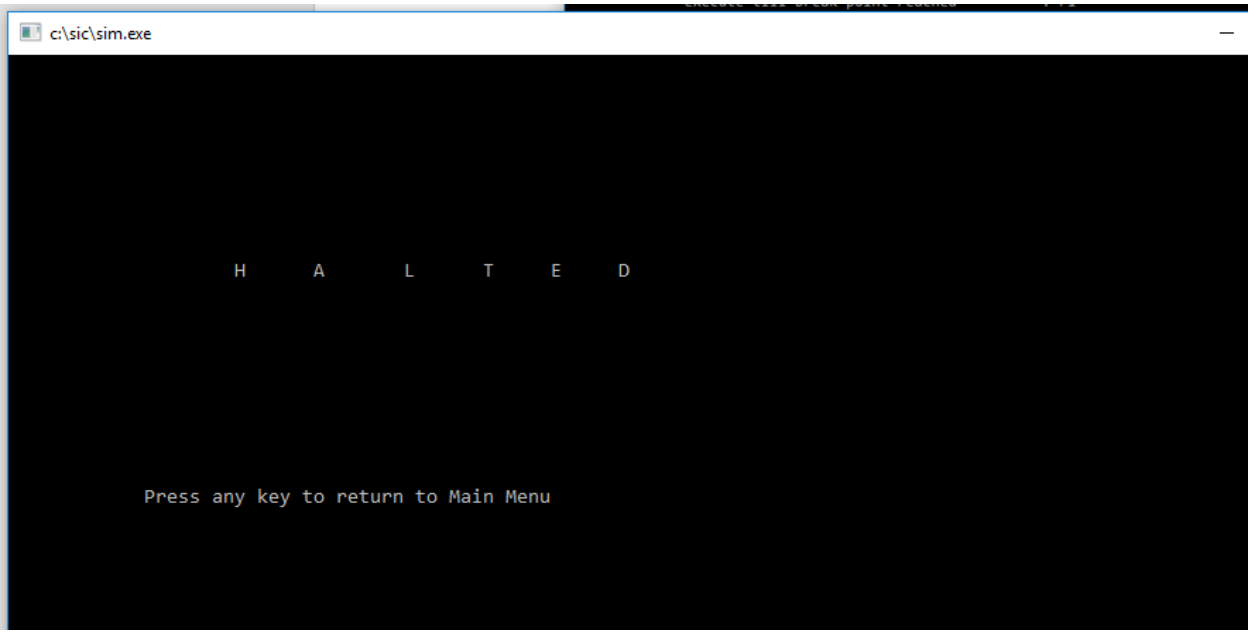
(before all examples).



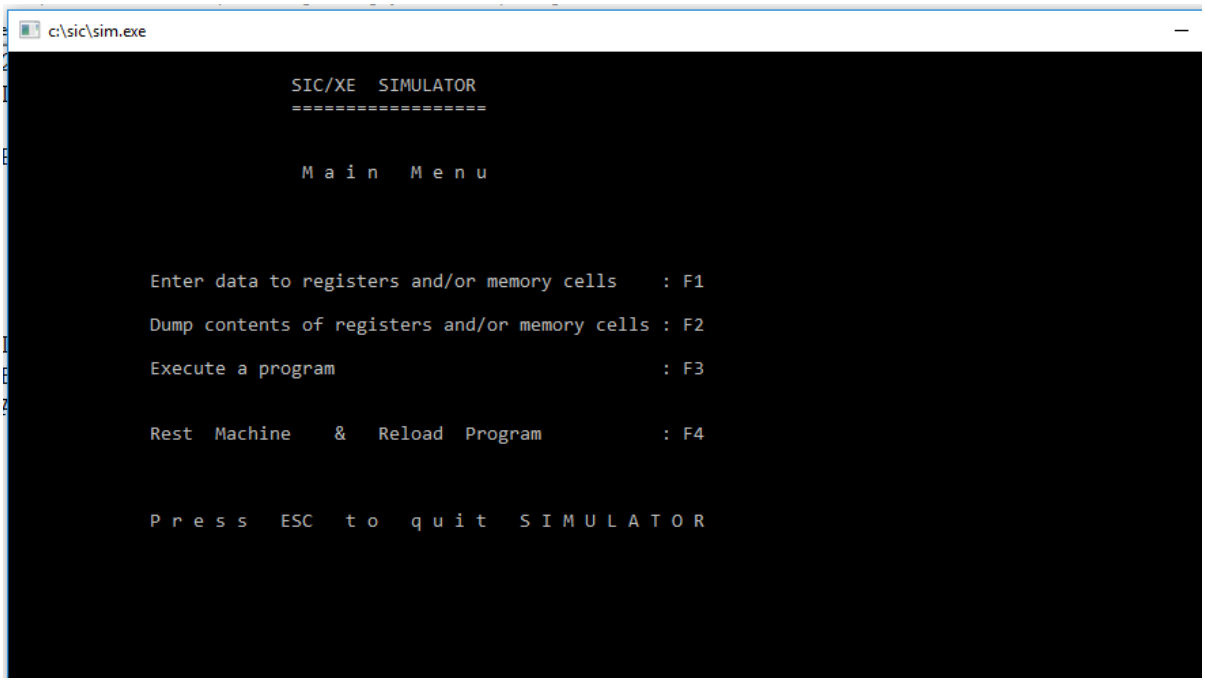
a. Execute a program (F3).



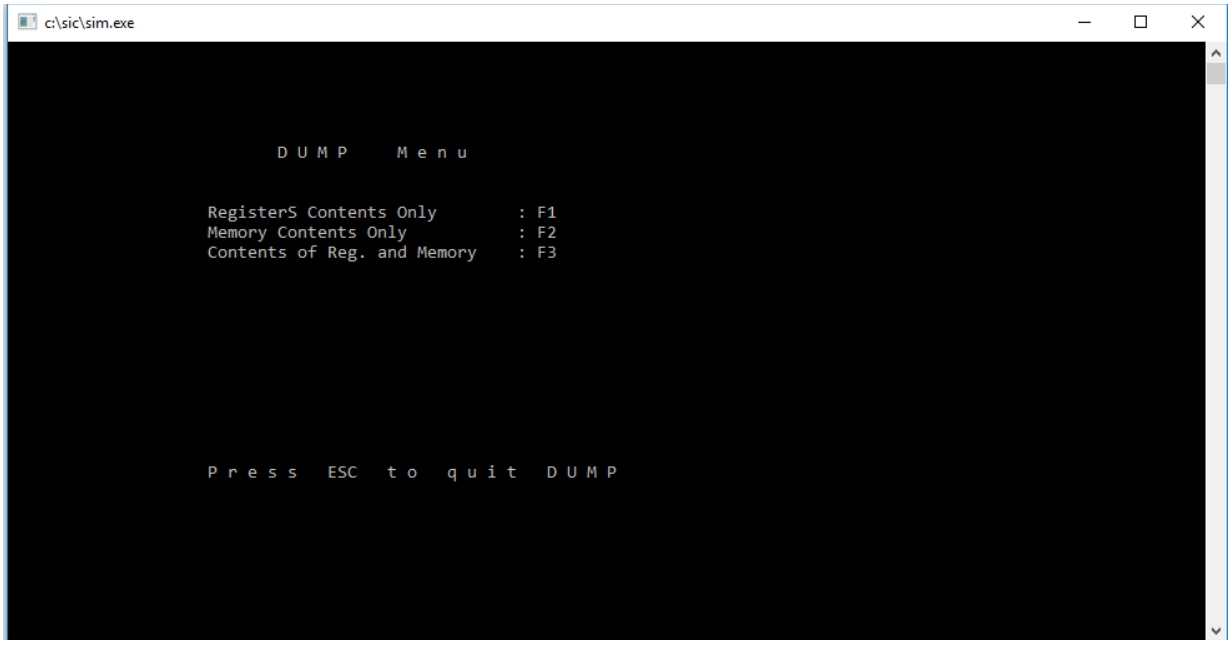
b. Automatic execution till halted (F4): And press any button to execute instruction till the end.



C. Dump contents of registers and/or memory cells (F2):

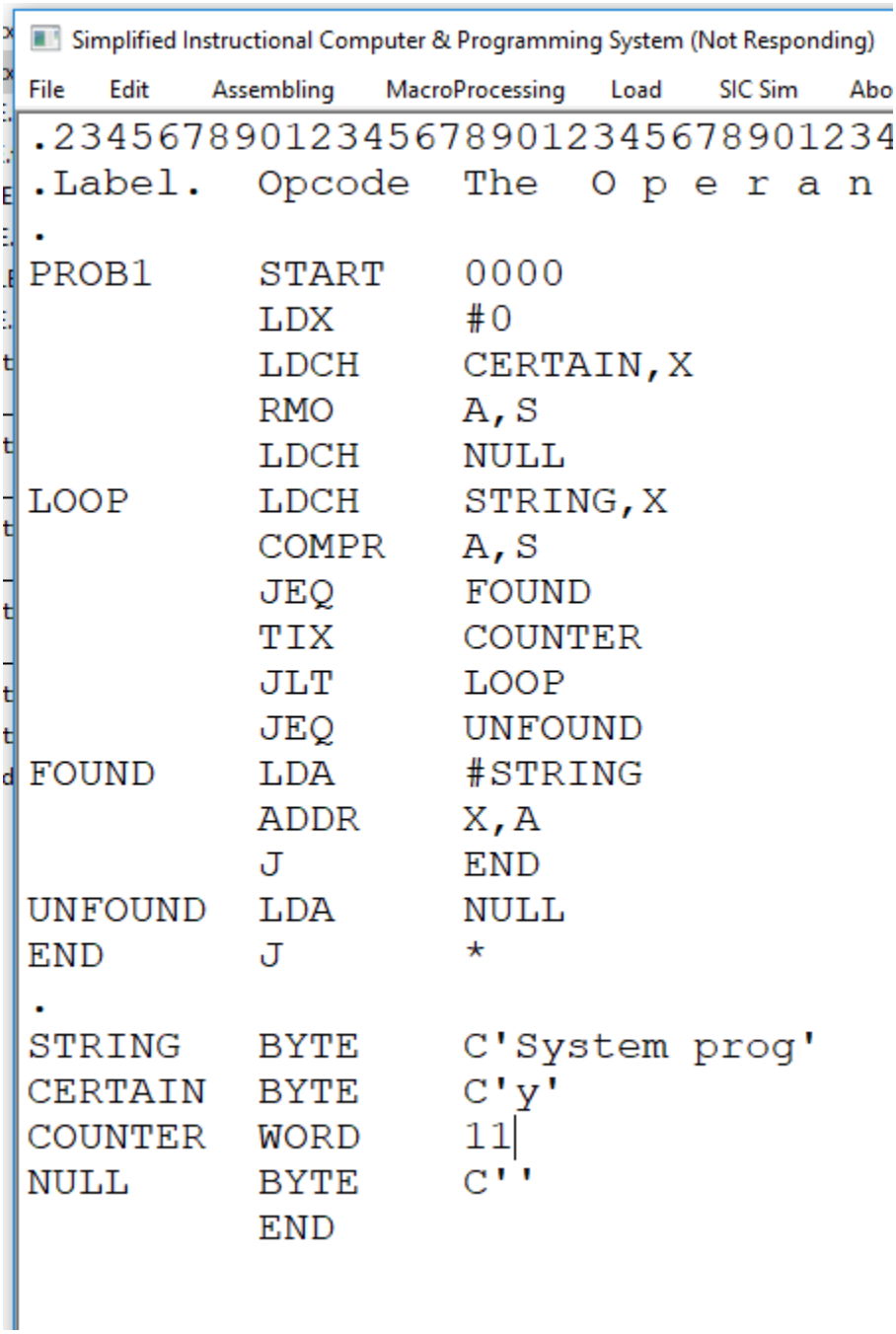


D. Contents of Reg. and Memory (F3):

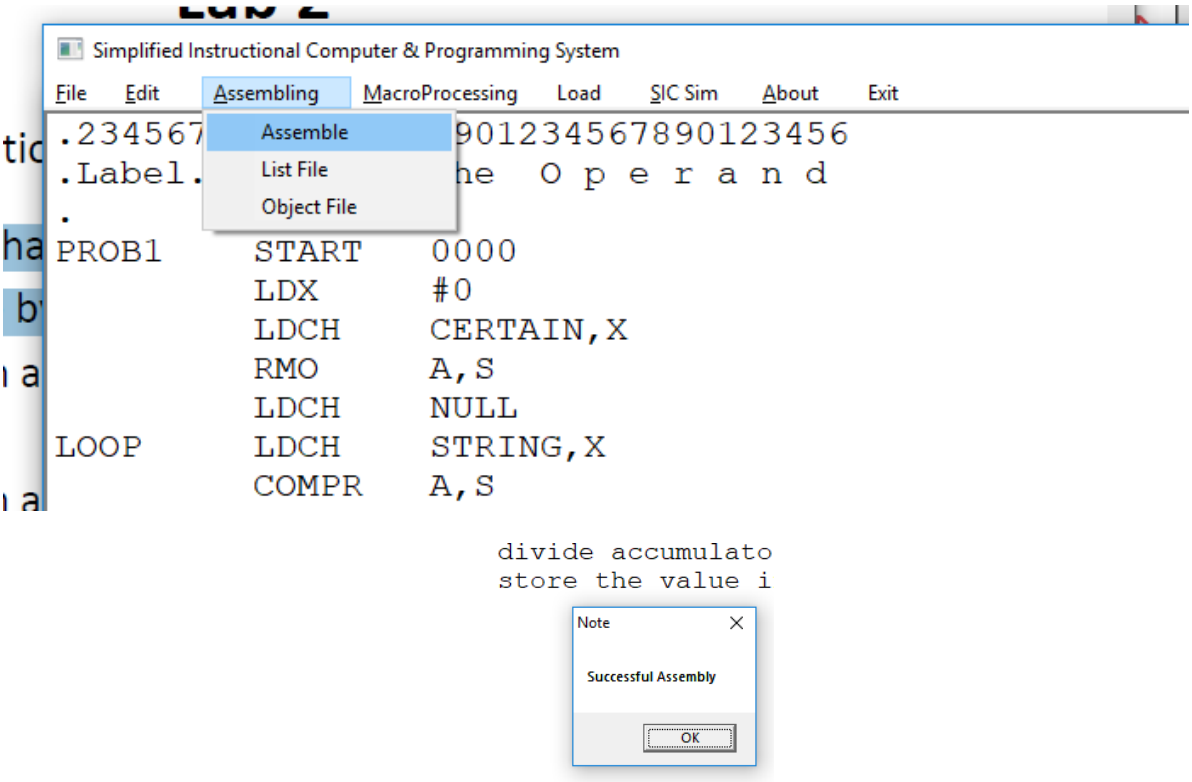


a. Write a program that searches for a certain byte in a string, if found, put the address of the byte in register A else, put 0xFFFFF:

1) Code:

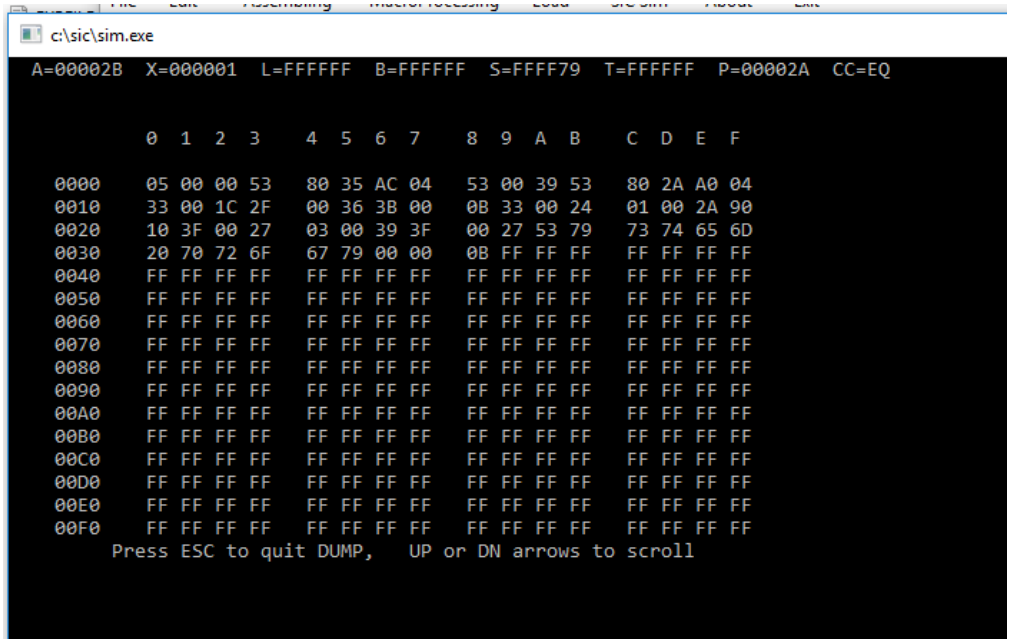


2) Assembling:

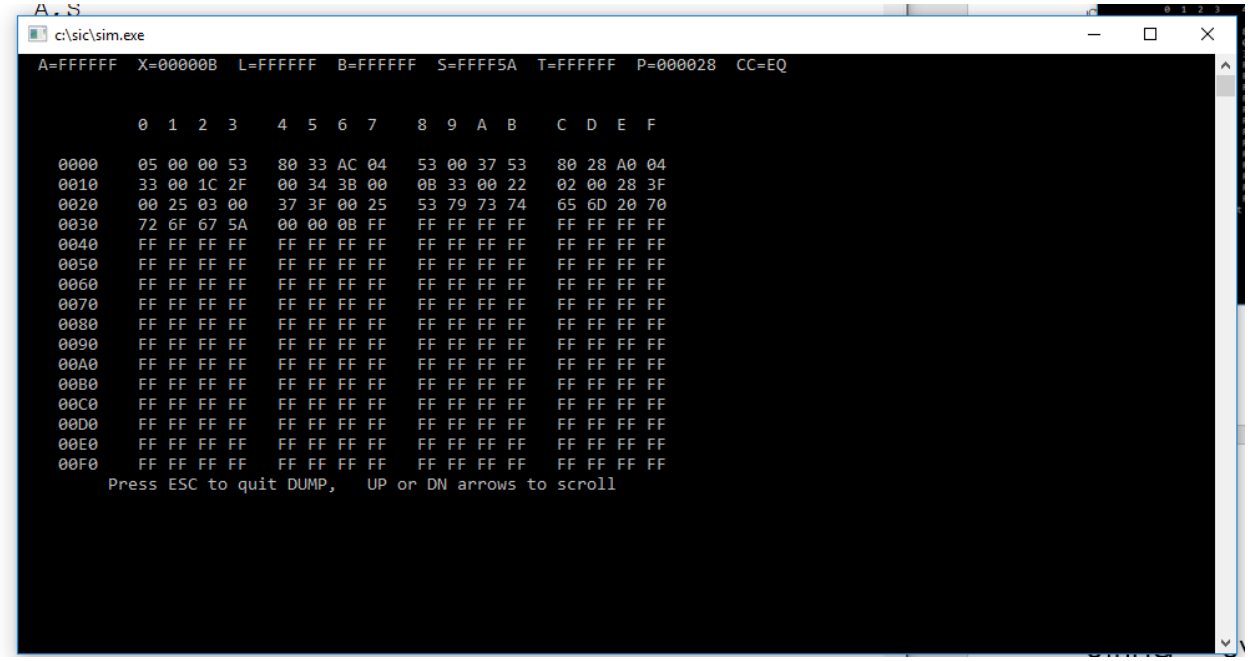


3) Sample run:

String = "System prog ", Certain = 'y',

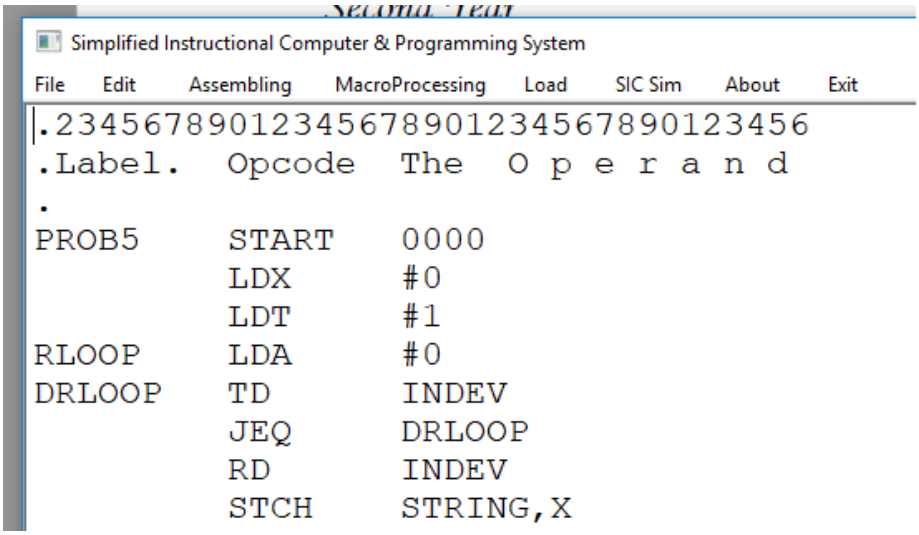


String = "System prog", Certain = 'Z',



b. Read a string from an input device and print it reversed to an output device:

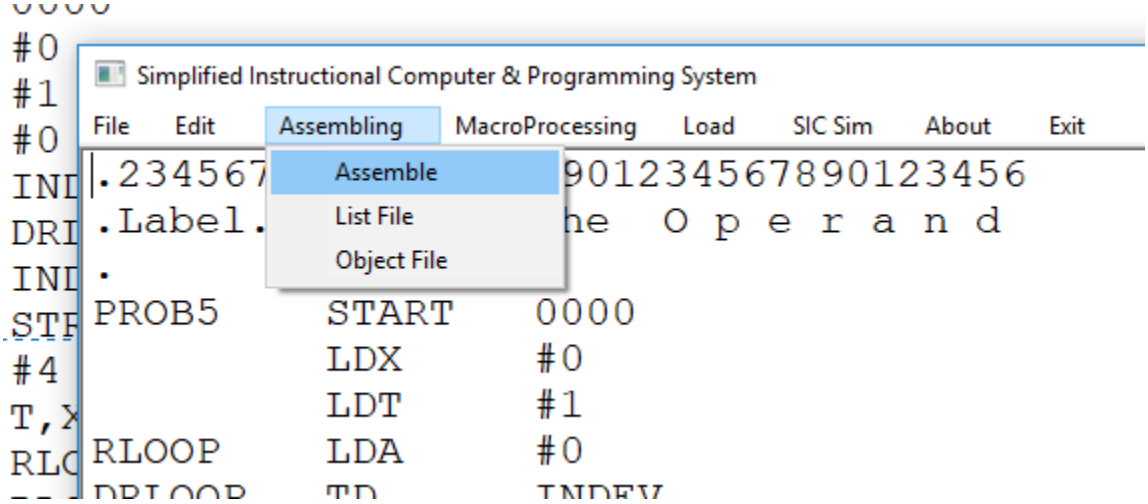
1) Code:



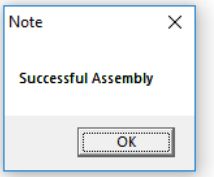
```
            COMP    #4
            ADDR    T,X
            JGT     RLOOP
            JLT     RLOOP
            SUBR    T,X
            SUBR    T,X
            LDS     #0
WLOOP       LDA     #0
DWLOOP      TD      OUTDEV
            JEQ     DWLOOP
            LDCH    STRING,X
            WD      OUTDEV
            SUBR    T,X
            COMPR   S,X
            JLT     DWLOOP
            JEQ     DWLOOP
            J       *
.
INDEV       BYTE    X'F1'
OUTDEV      BYTE    X'04'
STRING      RESB    100
            END
```

C:\SIC\files\prog_2__2.txt

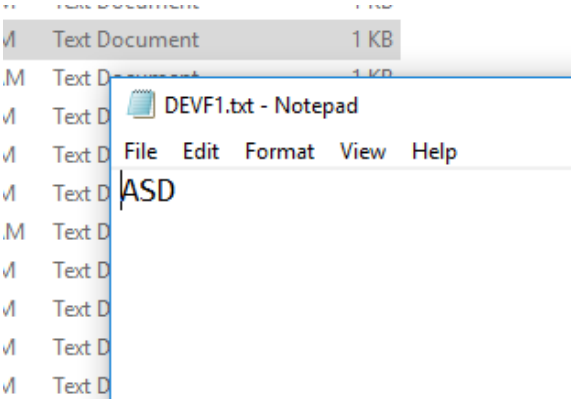
2) Assembling:

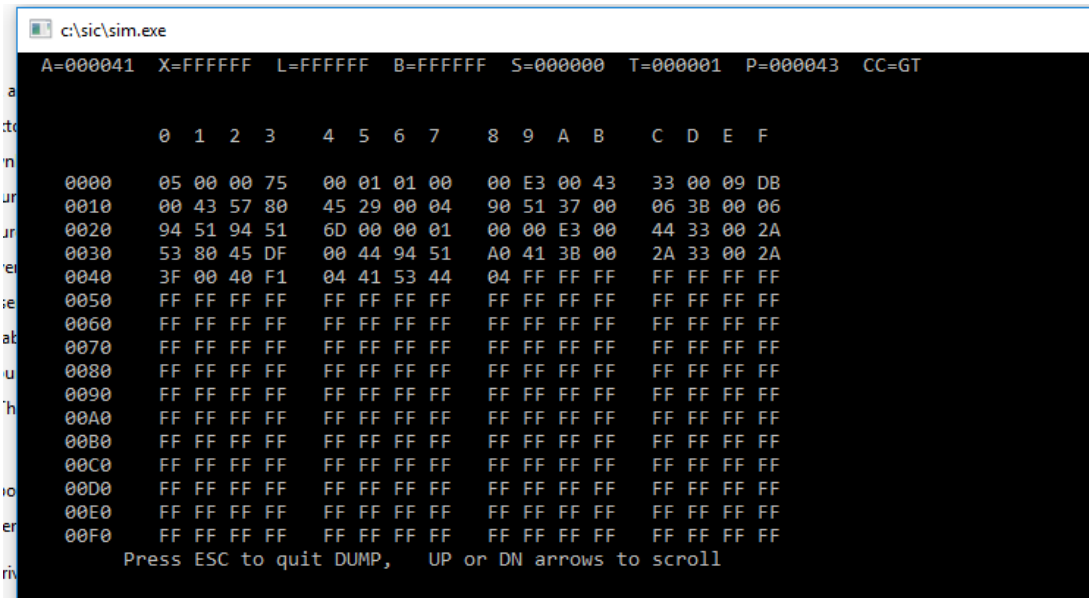


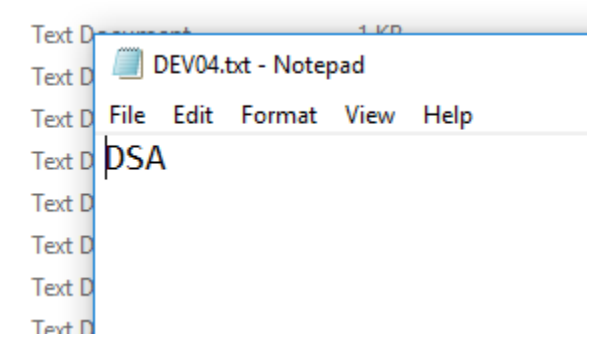
```
divide accumulato
store the value i
```

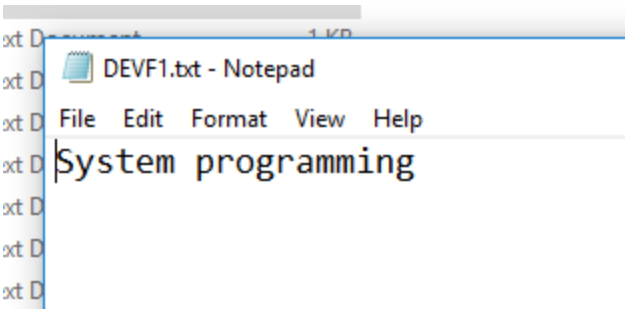


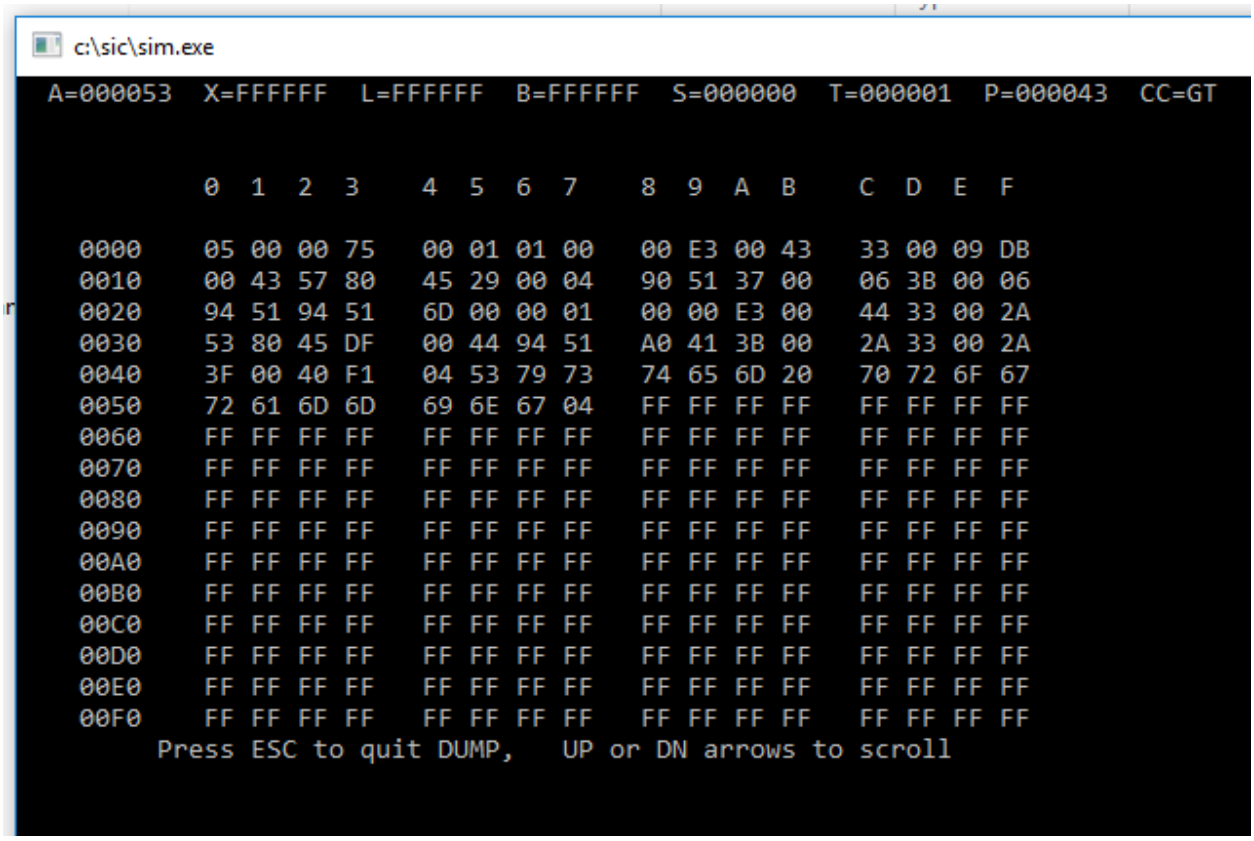
3) Sample run:

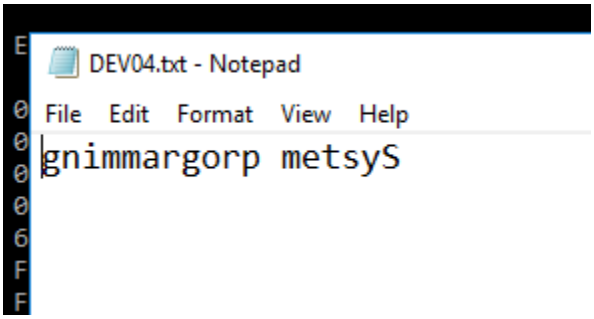






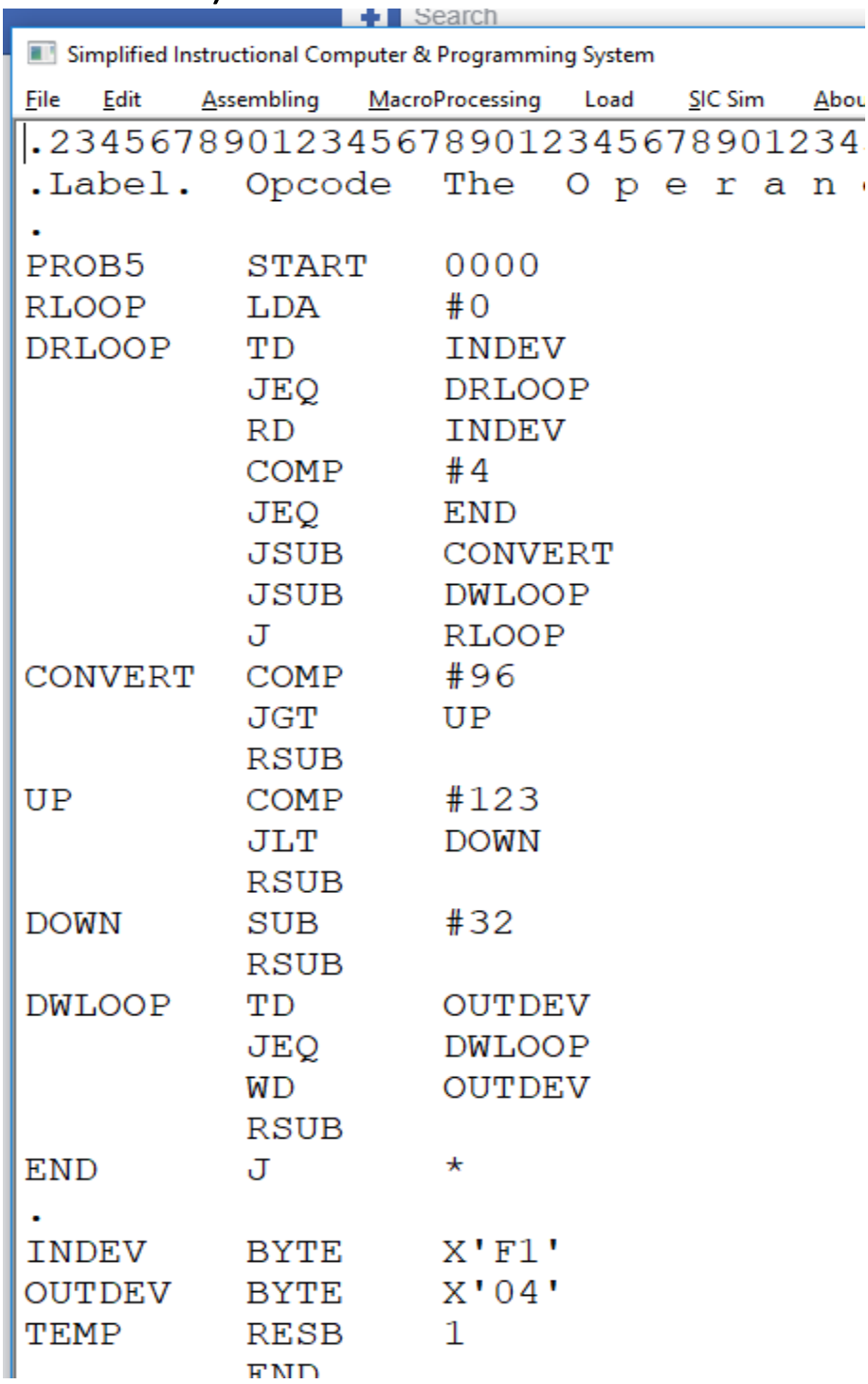

DEVF1.txt - Notepad
File Edit Format View Help
System programming


c:\sic\sim.exe
A=000053 X=FFFFFF L=FFFFFF B=FFFFFF S=000000 T=000001 P=000043 CC=GT
0 1 2 3 4 5 6 7 8 9 A B C D E F
0000 05 00 00 75 00 01 01 00 00 E3 00 43 33 00 09 DB
0010 00 43 57 80 45 29 00 04 90 51 37 00 06 3B 00 06
0020 94 51 94 51 6D 00 00 01 00 00 E3 00 44 33 00 2A
0030 53 80 45 DF 00 44 94 51 A0 41 3B 00 2A 33 00 2A
0040 3F 00 40 F1 04 53 79 73 74 65 6D 20 70 72 6F 67
0050 72 61 6D 6D 69 6E 67 04 FF FF FF FF FF FF FF FF
0060 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0070 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0080 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0090 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
00A0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
00B0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
00C0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
00D0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
00E0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
00F0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
Press ESC to quit DUMP, UP or DN arrows to scroll


DEV04.txt - Notepad
File Edit Format View Help
gnimmargorp metsyS

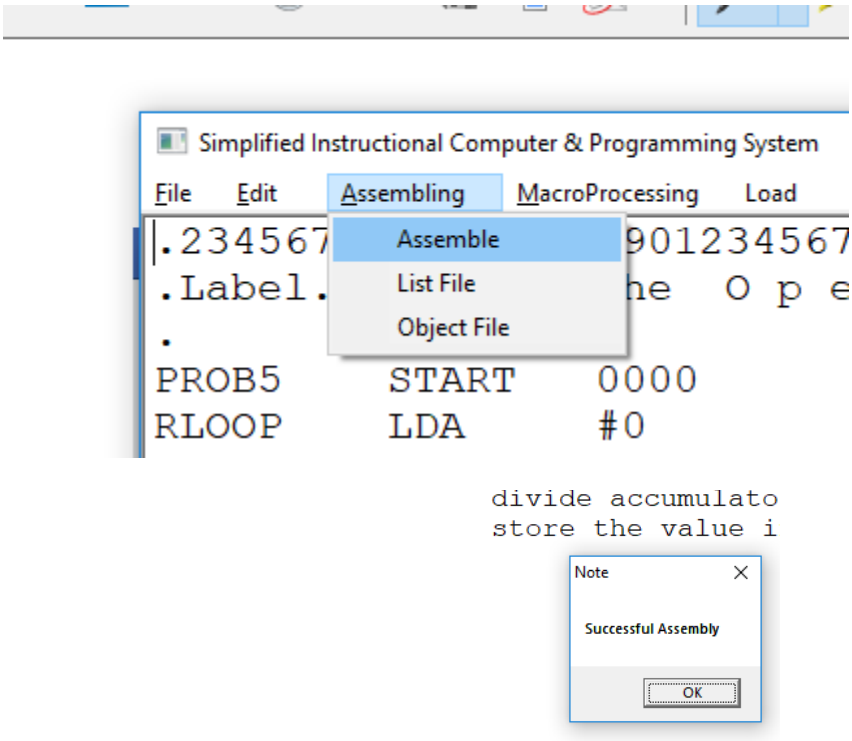
c. Read a string from an input device and print the string to an output device after converting it to UPPER case:

1) Code:

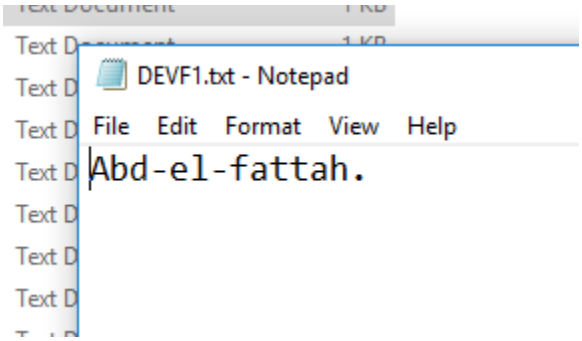


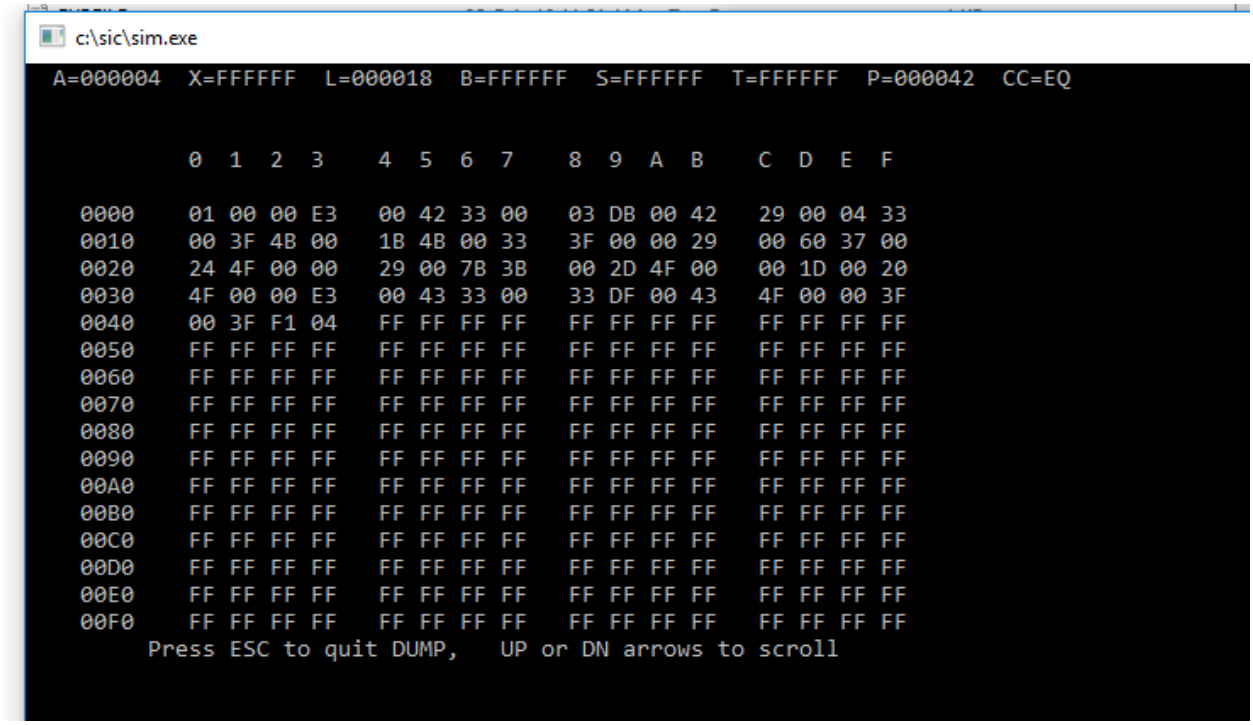

```
.2345678901234567890123456789012345
.Label.   Opcode   The   O p e r a n
.
PROB5     START    0000
RLOOP     LDA      #0
DRLOOP    TD       INDEV
          JEQ      DRLOOP
          RD       INDEV
          COMP     #4
          JEQ      END
          JSUB     CONVERT
          JSUB     DWLOOP
          J        RLOOP
CONVERT   COMP     #96
          JGT      UP
          RSUB
UP        COMP     #123
          JLT      DOWN
          RSUB
DOWN      SUB      #32
          RSUB
DWLOOP    TD       OUTDEV
          JEQ      DWLOOP
          WD       OUTDEV
          RSUB
END       J        *
.
INDEV     BYTE     X'F1'
OUTDEV    BYTE     X'04'
TEMP      RESB     1
          END
```

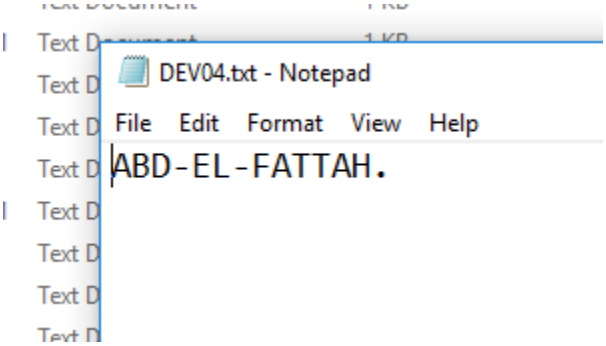
2) Assembling:



3) Sample run:

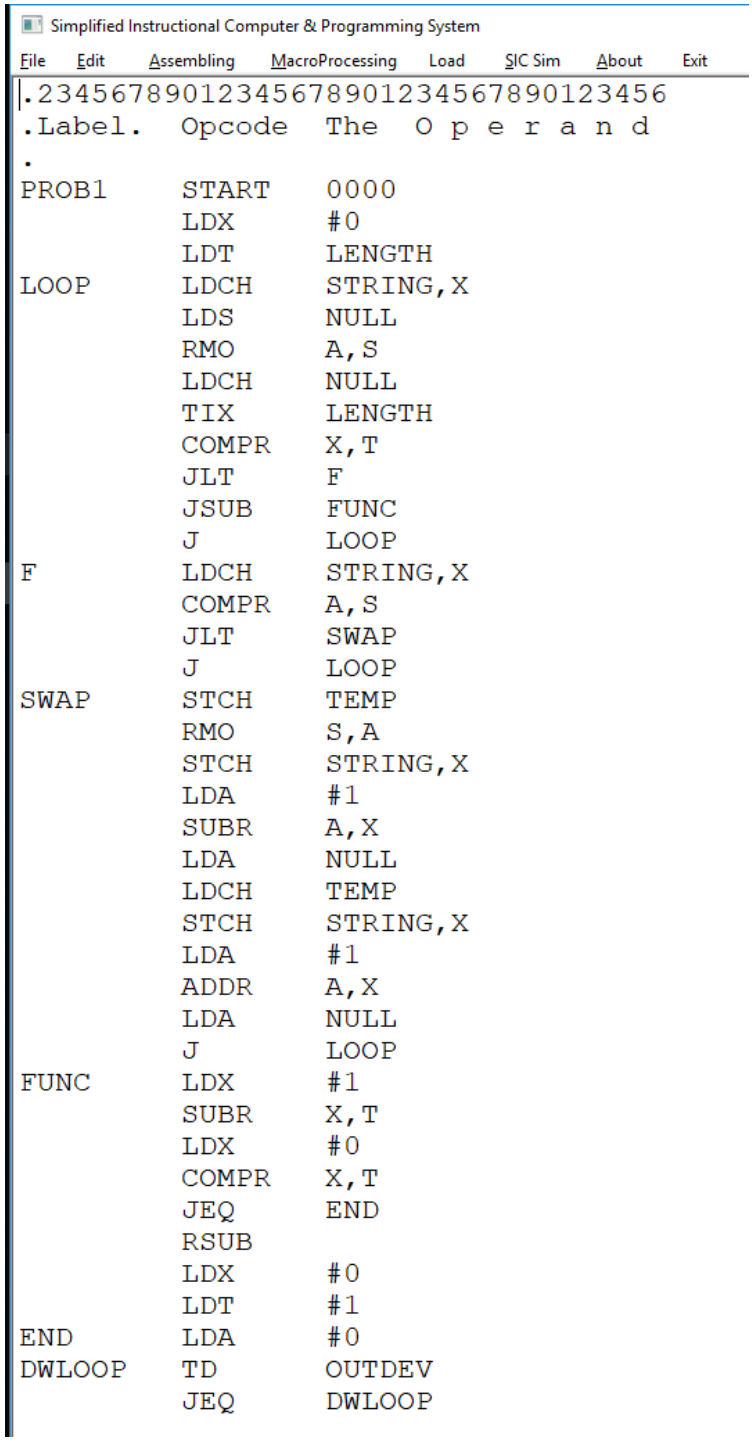



c:\sic\sim.exe
A=000004 X=FFFFFF L=000018 B=FFFFFF S=FFFFFF T=FFFFFF P=000042 CC=EQ
0 1 2 3 4 5 6 7 8 9 A B C D E F
0000 01 00 00 E3 00 42 33 00 03 DB 00 42 29 00 04 33
0010 00 3F 4B 00 1B 4B 00 33 3F 00 00 29 00 60 37 00
0020 24 4F 00 00 29 00 7B 3B 00 2D 4F 00 00 1D 00 20
0030 4F 00 00 E3 00 43 33 00 33 DF 00 43 4F 00 00 3F
0040 00 3F F1 04 FF FF FF FF FF FF FF FF FF FF FF FF
0050 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0060 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0070 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0080 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0090 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
00A0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
00B0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
00C0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
00D0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
00E0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
00F0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
Press ESC to quit DUMP, UP or DN arrows to scroll


DEV04.txt - Notepad
File Edit Format View Help
ABD-EL-FATTAH.

D. Implement the bubble sort to sort characters in a string:

1) Code:

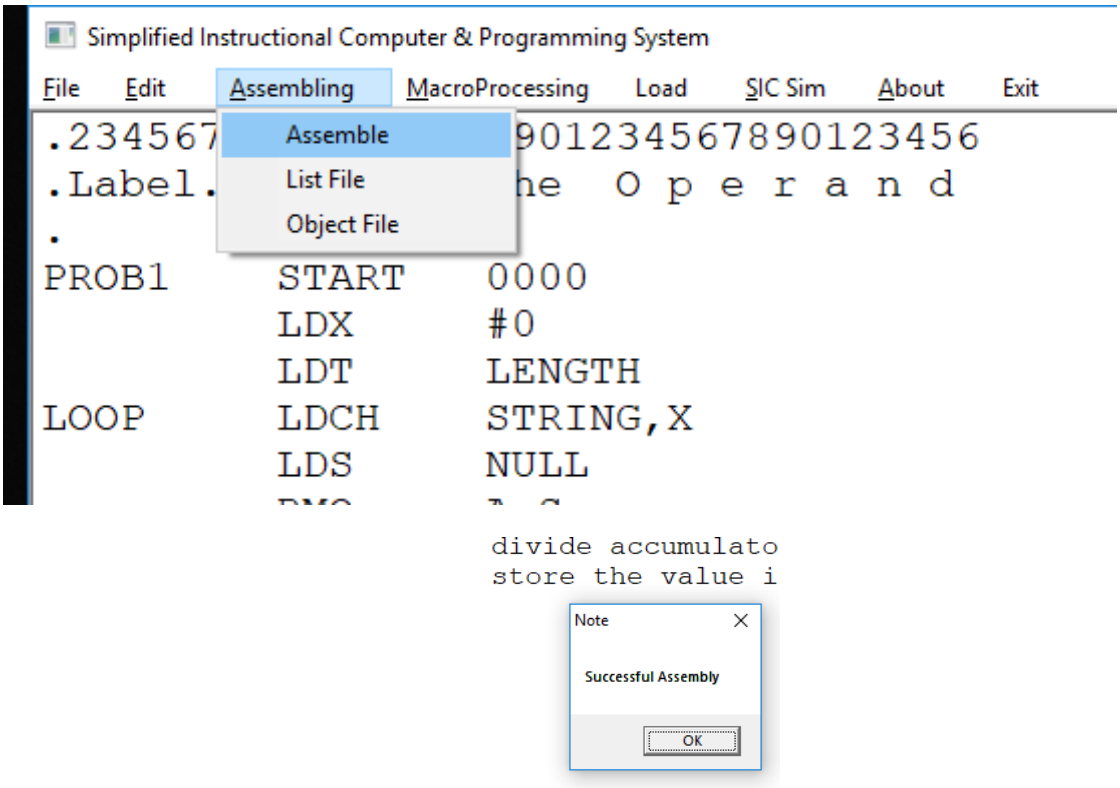



```
.23456789012345678901234567890123456
.Label.   Opcode   The   O p e r a n d
.
PROB1     START    0000
          LDX      #0
          LDT      LENGTH
LOOP      LDCH     STRING,X
          LDS      NULL
          RMO      A,S
          LDCH     NULL
          TIX      LENGTH
          COMPR    X,T
          JLT      F
          JSUB     FUNC
          J        LOOP
F         LDCH     STRING,X
          COMPR    A,S
          JLT      SWAP
          J        LOOP
SWAP      STCH     TEMP
          RMO      S,A
          STCH     STRING,X
          LDA      #1
          SUBR     A,X
          LDA      NULL
          LDCH     TEMP
          STCH     STRING,X
          LDA      #1
          ADDR     A,X
          LDA      NULL
          J        LOOP
FUNC      LDX      #1
          SUBR     X,T
          LDX      #0
          COMPR    X,T
          JEQ      END
          RSUB
          LDX      #0
          LDT      #1
END       LDA      #0
DWLOOP    TD       OUTDEV
          JEQ      DWLOOP
```

```
END       LDA      #0
DWLOOP    TD       OUTDEV
          JEQ      DWLOOP
          LDCH     STRING,X
          WD       OUTDEV
          TIX      LENGTH
          JLT      DWLOOP
          J        *
.
OUTDEV    BYTE     X'04'
STRING    BYTE     C'ECDBFA'
TEMP      RESB     1
LENGTH    WORD     6
NULL      BYTE     C''
          END
```

C:\SIC\files\prog_4__2.txt

2) Assembling:



3) Sample run:

String = "ECDABF " output = "ABCDEF"