ASCENDING ORDER

EXP NO: 12

AIM: To compute ascending order of an array using 8085 processor.

ALGORITHM:

1) Initialize HL pair as memory pointer.
2) Get the count at memory and load it into C register
3) Copy it in D register (for bubble sort (N-1)) times required).
4) Get the first value in A register.
5) Compare it with the value at next location.
6) If they are out of order, exchange the contents of A register and memory.
7) Decrement D register content by 1
8) Repeat step 5 and 7 till the value in D register become zero.
9) Decrement the C register content by 1.
10) Repeat steps 3 to 9 till the value in C register becomes zero.

PROGRAM:sssssssss

```
LXI H,8000
MOV C,M

DCR  C

LOOP3: MOV D,C
LXI H,8001
LOOP2: MOV A,M
INX H
CMP   M

JC LOOP1
MOV B,M

MOV M,A
DCX H
MOV M,B
INX H
LOOP1: DCR  D
JNZ   LOOP2

DCR  C
JNZ  LOOP3
HLT
```

INPUT:

4

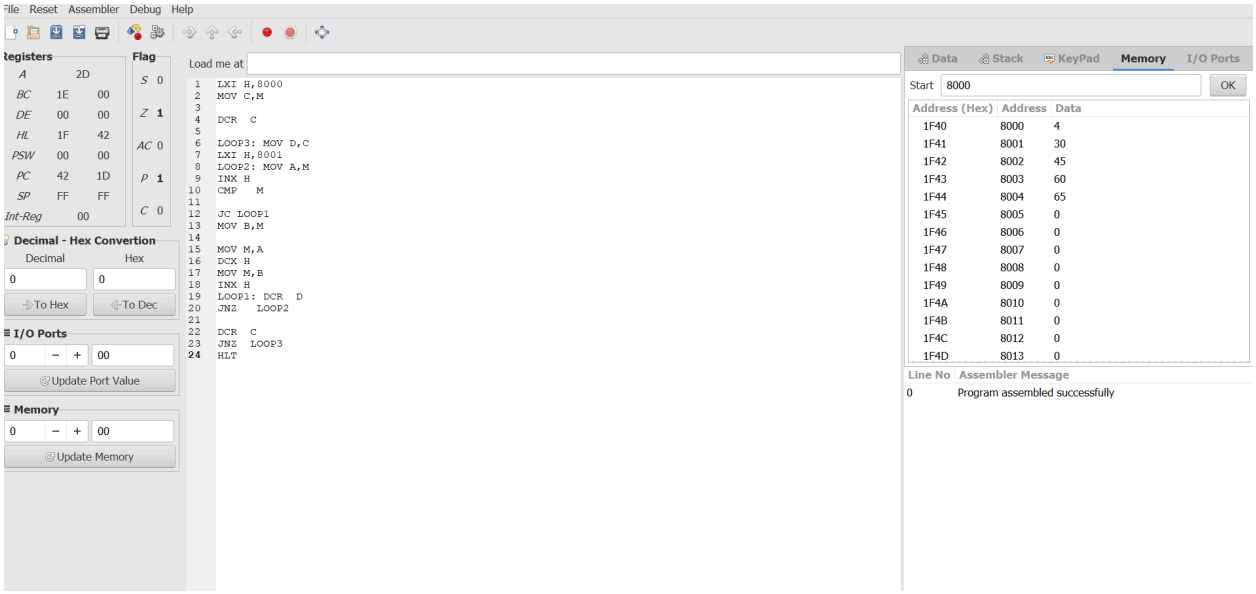
60

65

45

30

OUTPUT:


File Reset Assembler Debug Help
Registers
A 2D
BC 1E 00
DE 00 00
HL 1F 42
PSW 00 00
PC 42 1D
SP FF FF
Int-Reg 00
Flag
S 0
Z 1
AC 0
P 1
C 0
Decimal - Hex Convertion
Decimal
Hex
0
0
To Hex
To Dec
I/O Ports
0 00
Update Port Value
Memory
0 00
Update Memory
Load me at
1 LXI H,8000
2 MOV C,M
3
4 DCR C
5
6 LOOP3: MOV D,C
7 LXI H,8001
8 LOOP2: MOV A,M
9 INX H
10 CMP M
11
12 JC LOOP1
13 MOV B,M
14
15 MOV M,A
16 DCX H
17 MOV M,B
18 INX H
19 LOOP1: DCR D
20 JNZ LOOP2
21
22 DCR C
23 JNZ LOOP3
24 HLT
Data
Stack
KeyPad
Memory
I/O Ports
Start 8000
OK
Address (Hex) Address Data
1F40 8000 4
1F41 8001 30
1F42 8002 45
1F43 8003 60
1F44 8004 65
1F45 8005 0
1F46 8006 0
1F47 8007 0
1F48 8008 0
1F49 8009 0
1F4A 8010 0
1F4B 8011 0
1F4C 8012 0
1F4D 8013 0
Line No Assembler Message
0 Program assembled successfully