MIT Assignment-04

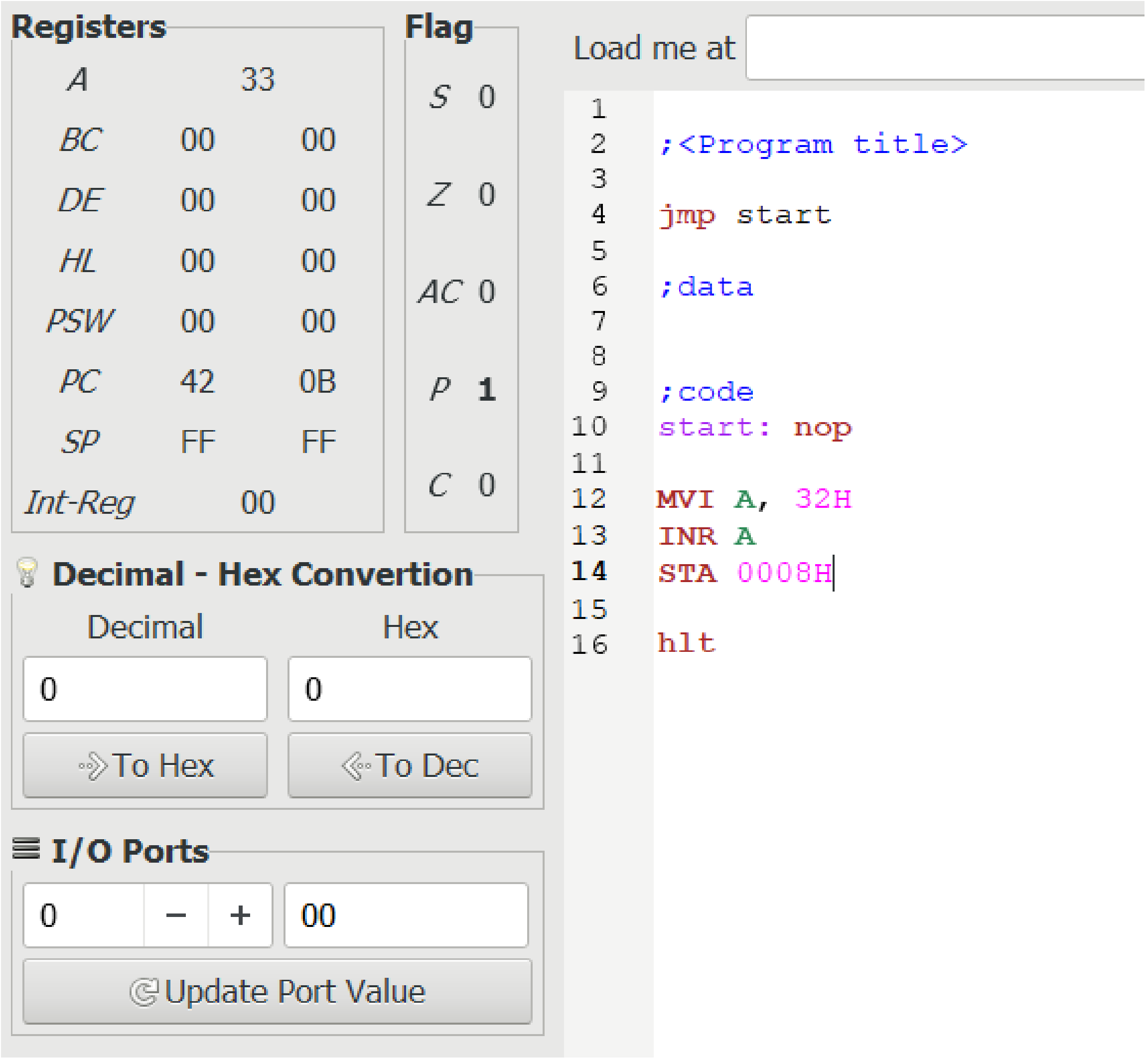
Jeetu Mehra

U24AI032

Question 1 : **Write instruction to increment the contents of a register and show result to memory location**

0008H.

Output:



Registers

A	33	
BC	00	00
DE	00	00
HL	00	00
PSW	00	00
PC	42	0B
SP	FF	FF
Int-Reg	00	

Flag

S	0
Z	0
AC	0
P	1
C	0

Load me at

```
;<Program title>

jmp start

;data


;code
start: nop

MVI A, 32H
INR A
STA 0008H

hlt
```

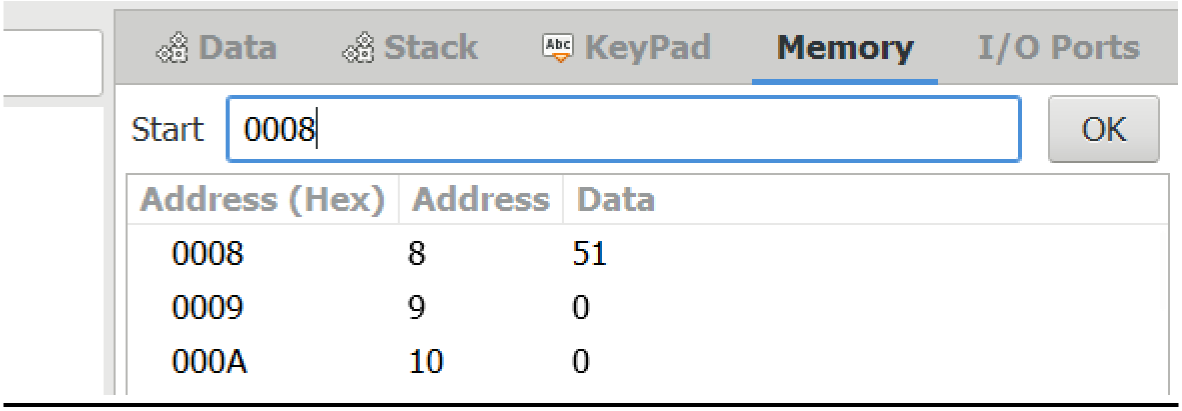
Decimal - Hex Convertion

Decimal: 0

Hex: 0

I/O Ports

0 — + 00

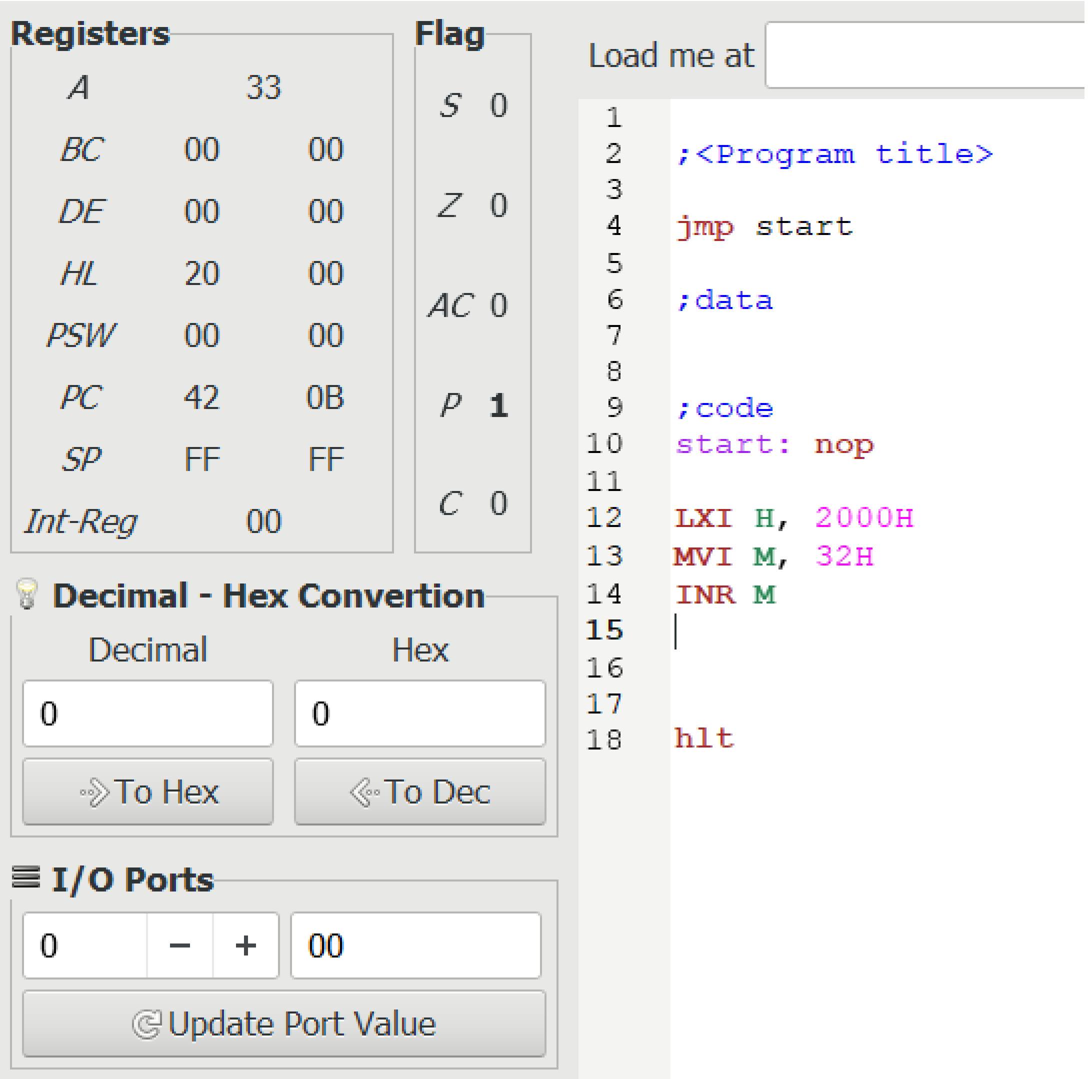


Start 0008

Address (Hex)	Address	Data
0008	8	51
0009	9	0
000A	10	0

Question 2: **Write instruction to increment the contents of a memory location the address of which is in HL Pair.**

Output:



Registers

A	33	
BC	00	00
DE	00	00
HL	20	00
PSW	00	00
PC	42	0B
SP	FF	FF
Int-Reg	00	

Flag

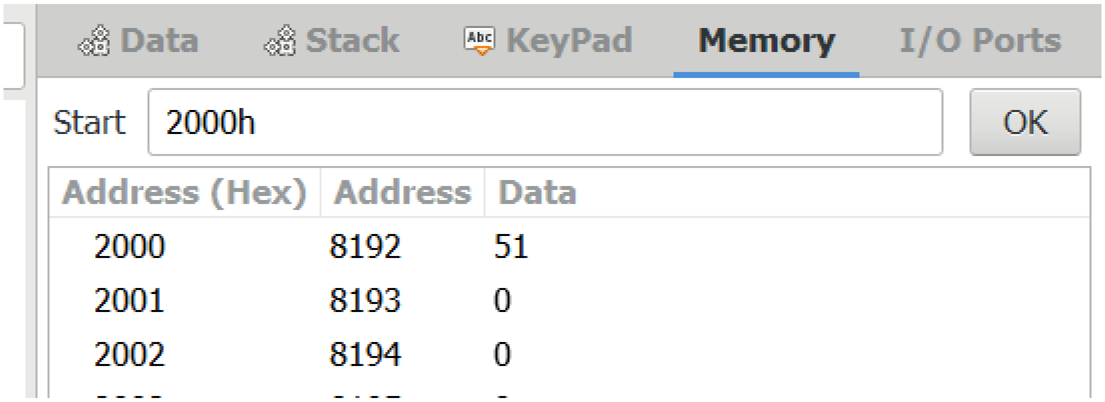
S	0
Z	0
AC	0
P	1
C	0

Decimal 0 Hex 0

I/O Ports 0 00

Load me at

```

;<Program title>

jmp start

;data


;code
start: nop

LXI H, 2000H
MVI M, 32H
INR M



hlt
```

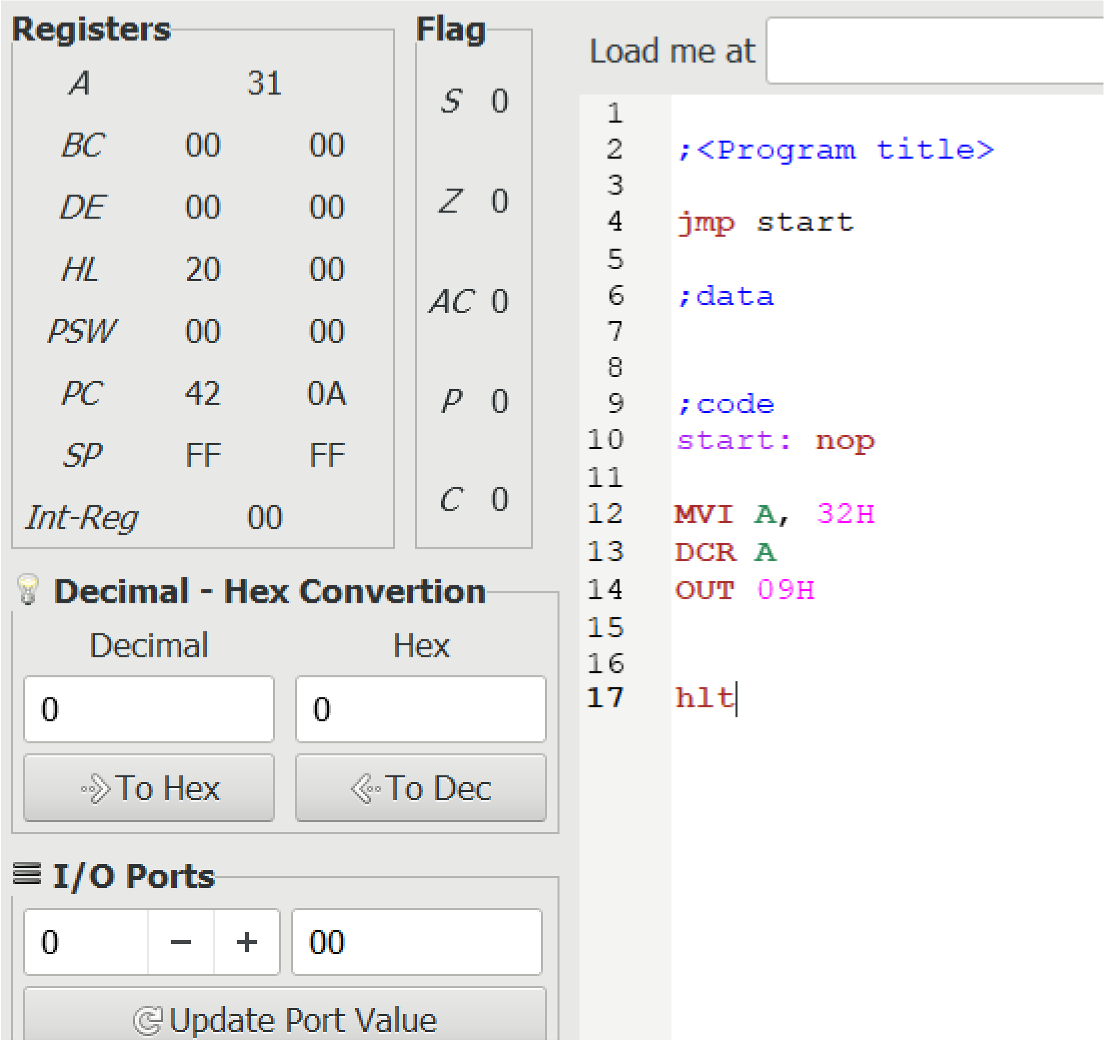


Start 2000h

Address (Hex)	Address	Data
2000	8192	51
2001	8193	0
2002	8194	0

Question 3: **Write instruction to decrement the contents of a register and show result to I/O port address 0009H.**

Output:



Registers		
A	31	
BC	00	00
DE	00	00
HL	20	00
PSW	00	00
PC	42	0A
SP	FF	FF
Int-Reg	00	

Flag	
S	0
Z	0
AC	0
P	0
C	0

Decimal 0 Hex 0

I/O Ports 0 00

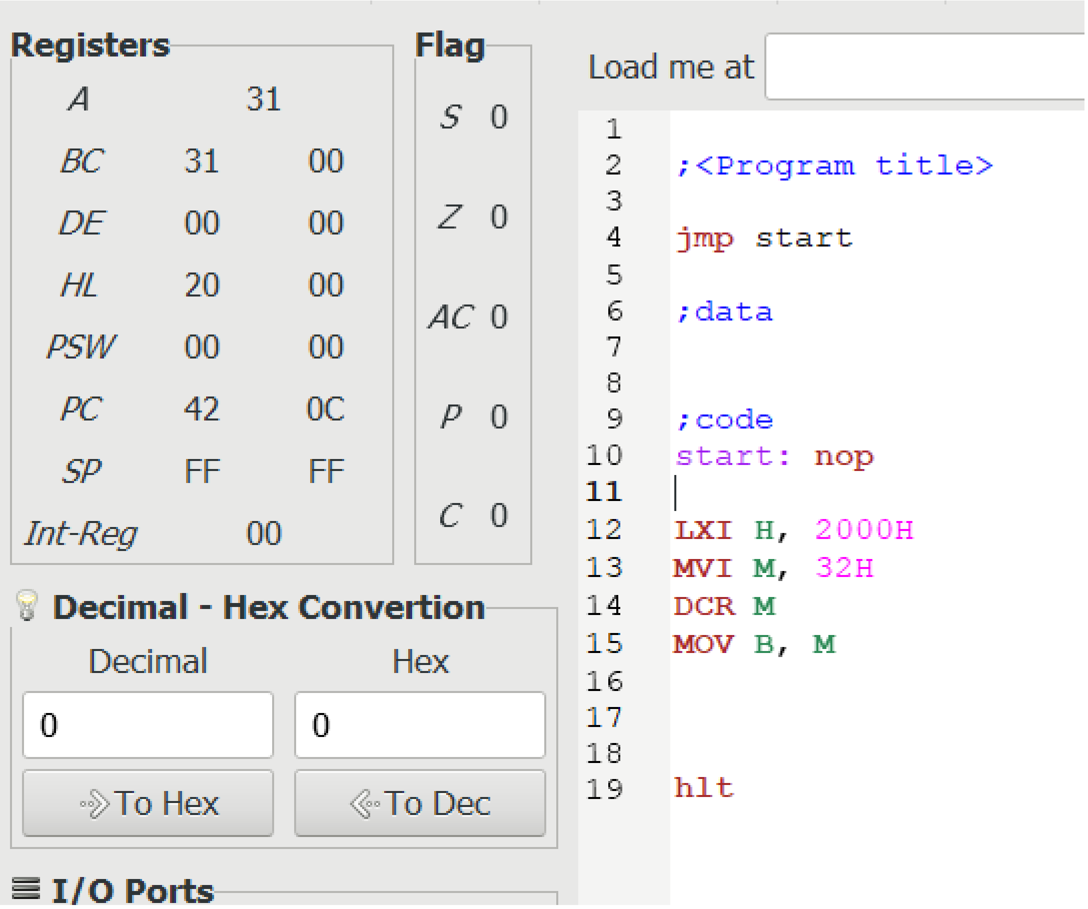
Load me at

```

;<Program title>

jmp start

;data


;code
start: nop

MVI A, 32H
DCR A
OUT 09H


hlt
```

<u>Question 4</u> : **Write instruction to decrement the contents of a memory location the address of which is in HL**

pair and copy that result to register B.

<u>Output:</u>



Registers

A	31	
BC	31	00
DE	00	00
HL	20	00
PSW	00	00
PC	42	0C
SP	FF	FF
Int-Reg	00	

Flag

S	0
Z	0
AC	0
P	0
C	0

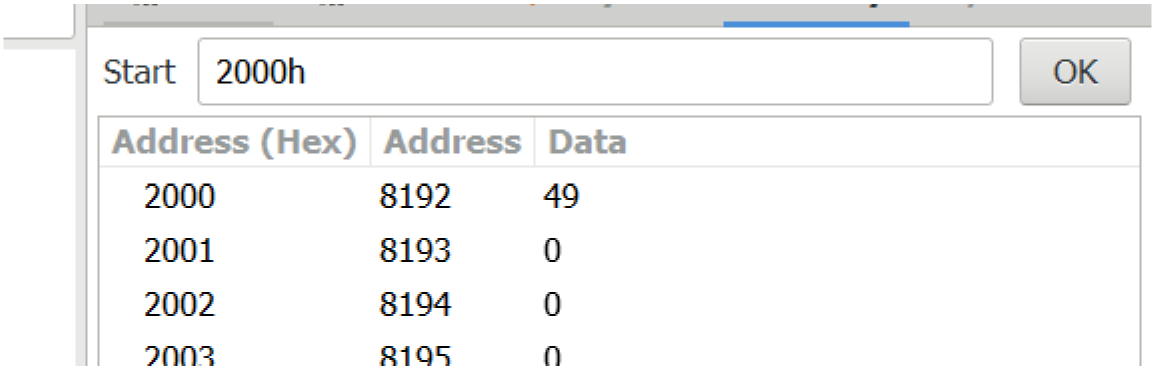
Decimal - Hex Convertion

Decimal: 0

Hex: 0

Load me at

```

;<Program title>

jmp start

;data


;code
start: nop

LXI H, 2000H
MVI M, 32H
DCR M
MOV B, M



hlt
```

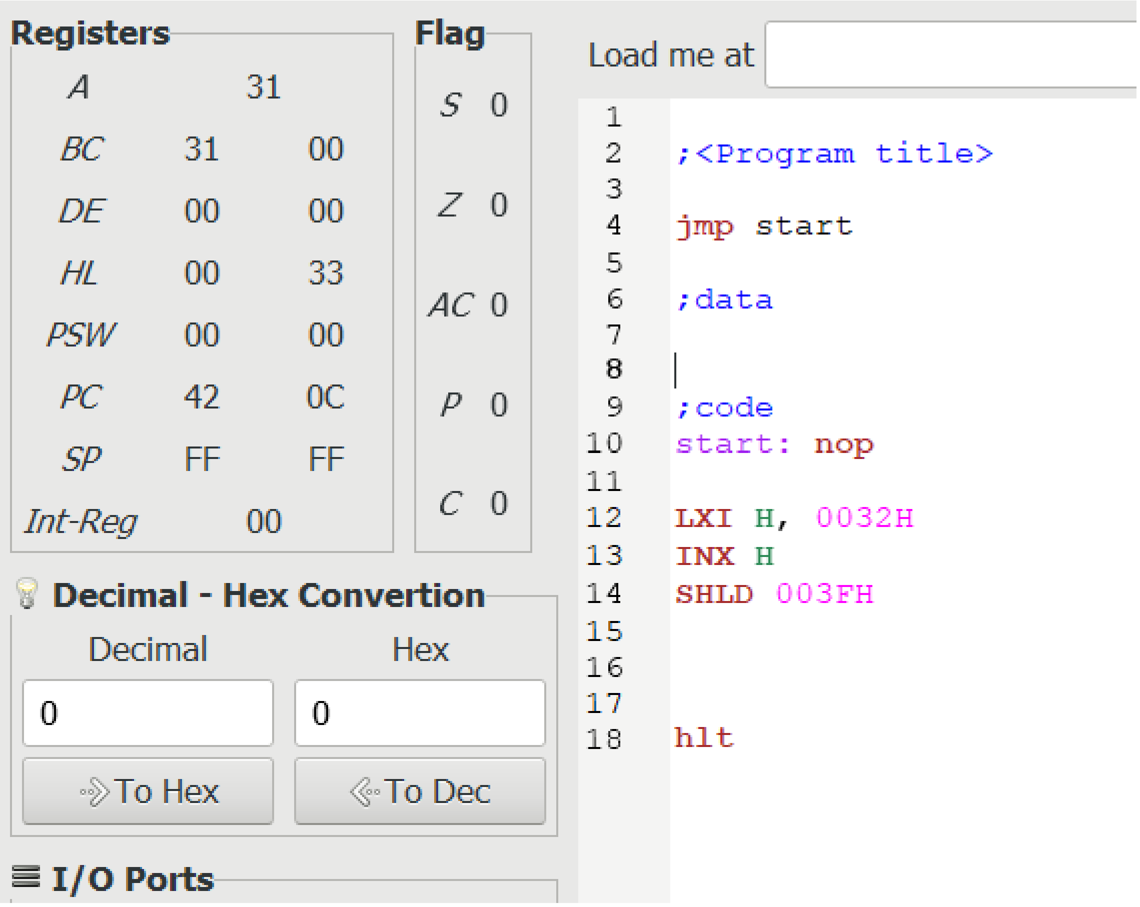


Start: 2000h

Address (Hex)	Address	Data
2000	8192	49
2001	8193	0
2002	8194	0
2003	8195	0

<u>Question 5</u> : Write instruction to increment the contents of a register-pair and show result to memory location 003FH.

<u>Output:</u>



Registers

A	31	
BC	31	00
DE	00	00
HL	00	33
PSW	00	00
PC	42	0C
SP	FF	FF
Int-Reg	00	

Flag

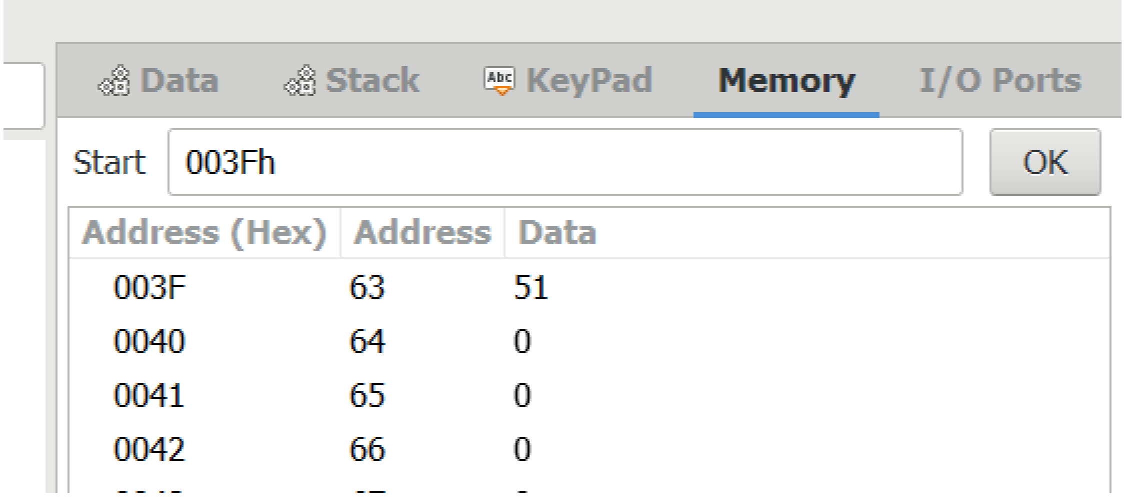
S	0
Z	0
AC	0
P	0
C	0

Decimal - Hex Convertion

Decimal: 0

Hex: 0

```

;<Program title>

jmp start

;data


;code
start: nop

LXI H, 0032H
INX H
SHLD 003FH



hlt
```

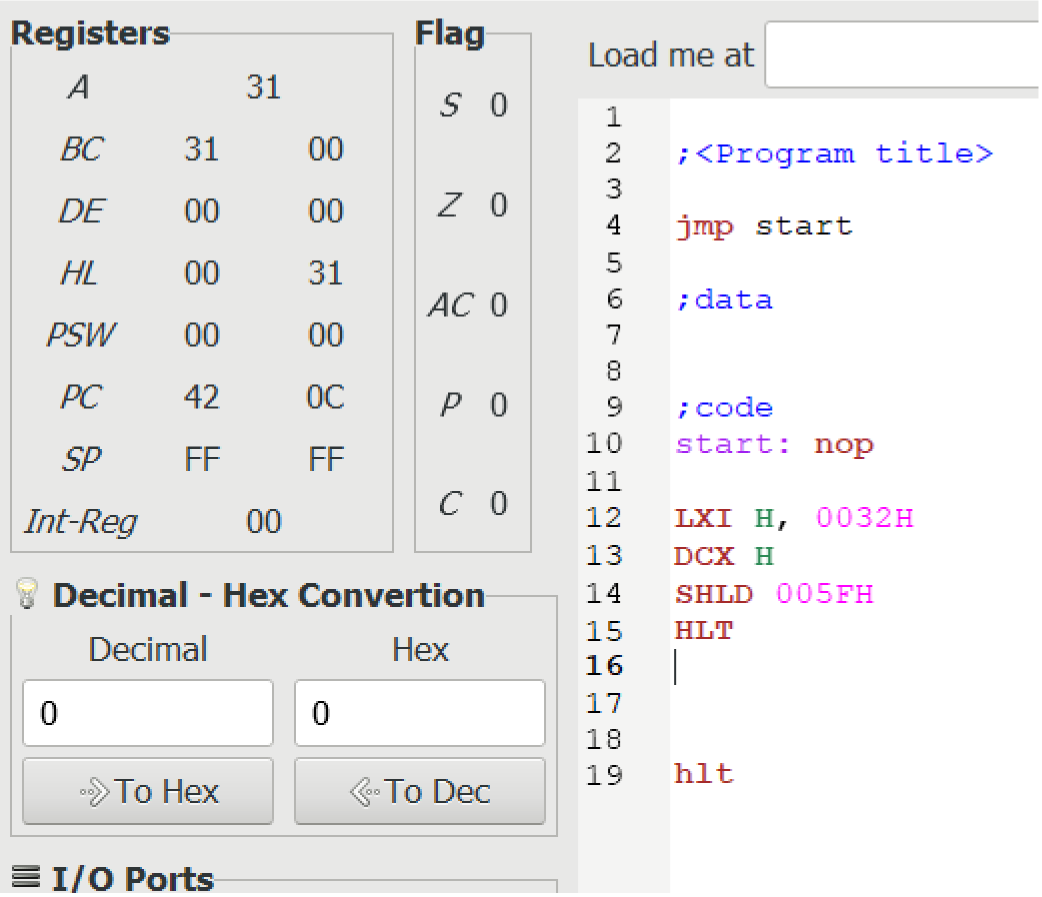


Start 003Fh

Address (Hex)	Address	Data
003F	63	51
0040	64	0
0041	65	0
0042	66	0

Question 6 : **Write instruction to decrement the contents of a register-pair and show result to memory location 005FH.**

Output:



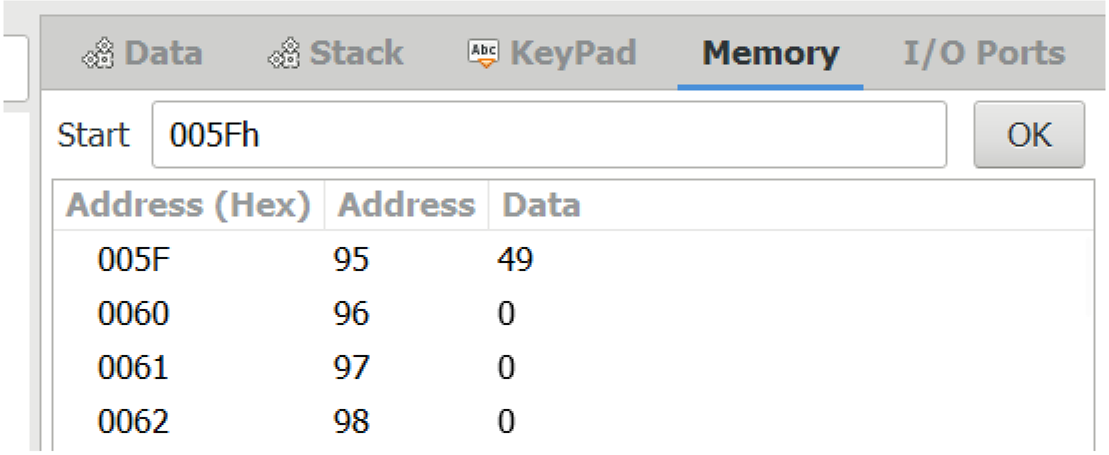
Registers

A	31	
BC	31	00
DE	00	00
HL	00	31
PSW	00	00
PC	42	0C
SP	FF	FF
Int-Reg	00	

Flag

S	0
Z	0
AC	0
P	0
C	0

```

;<Program title>

jmp start

;data


;code
start: nop

LXI H, 0032H
DCX H
SHLD 005FH
HLT



hlt
```



Start 005Fh

Address (Hex)	Address	Data
005F	95	49
0060	96	0
0061	97	0
0062	98	0

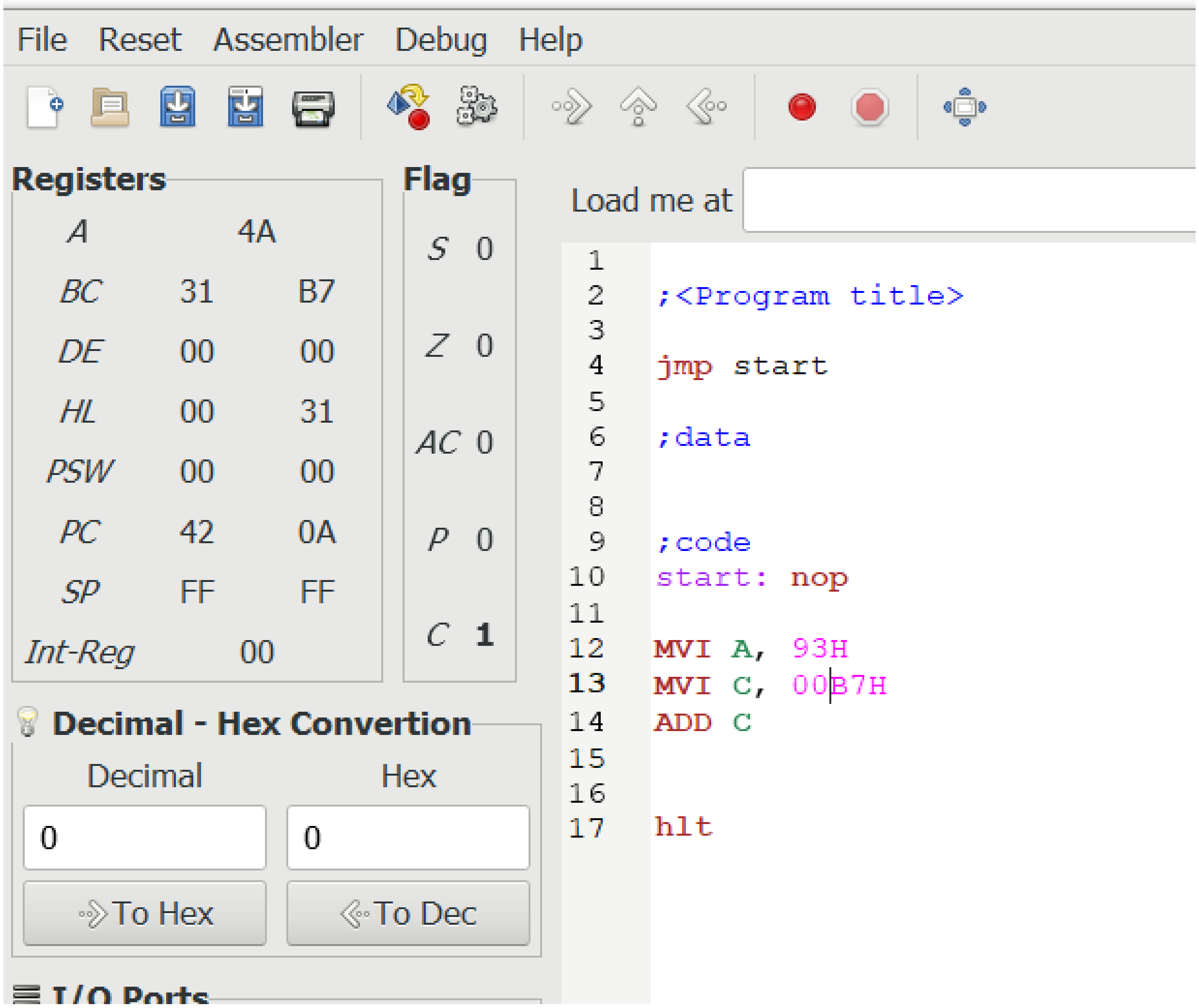
Question 7 : **Write instruction to do following operations:**

a. Load 93H in Accumulator.

b. Load B7 H in register C.

c. Addition of the above two numbers.

<u>Output:</u>



Registers		
A	4A	
BC	31	B7
DE	00	00
HL	00	31
PSW	00	00
PC	42	0A
SP	FF	FF
Int-Reg	00	

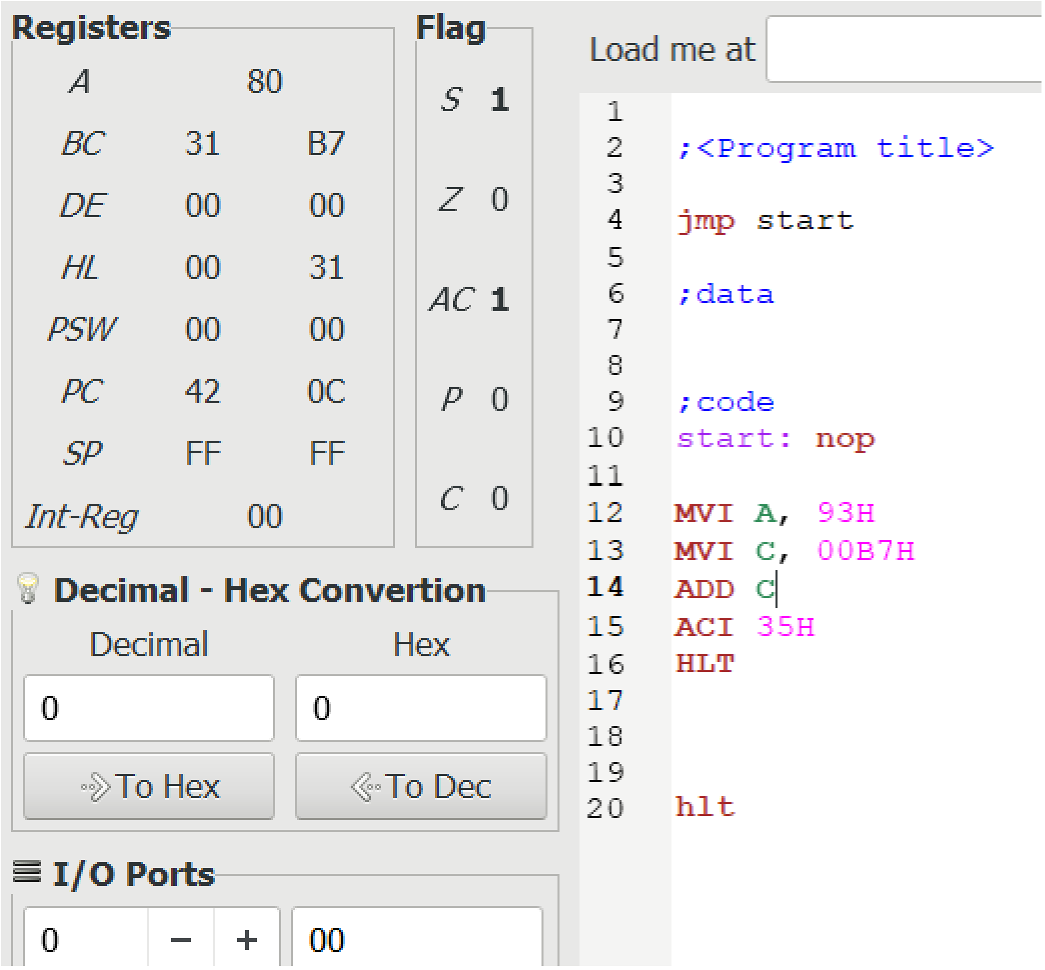
Flag	
S	0
Z	0
AC	0
P	0
C	1

Decimal	Hex
0	0

```

;<Program title>

jmp start

;data


;code
start: nop

MVI A, 93H
MVI C, 00B7H
ADD C


hlt
```

<u>Question 8</u> : Write instruction to add the 35H to the sum in the previous example when the carry flag is set.

<u>Output:</u>



Registers

A	80	
BC	31	B7
DE	00	00
HL	00	31
PSW	00	00
PC	42	0C
SP	FF	FF
Int-Reg	00	

Flag

S	1
Z	0
AC	1
P	0
C	0

Decimal: 0 Hex: 0

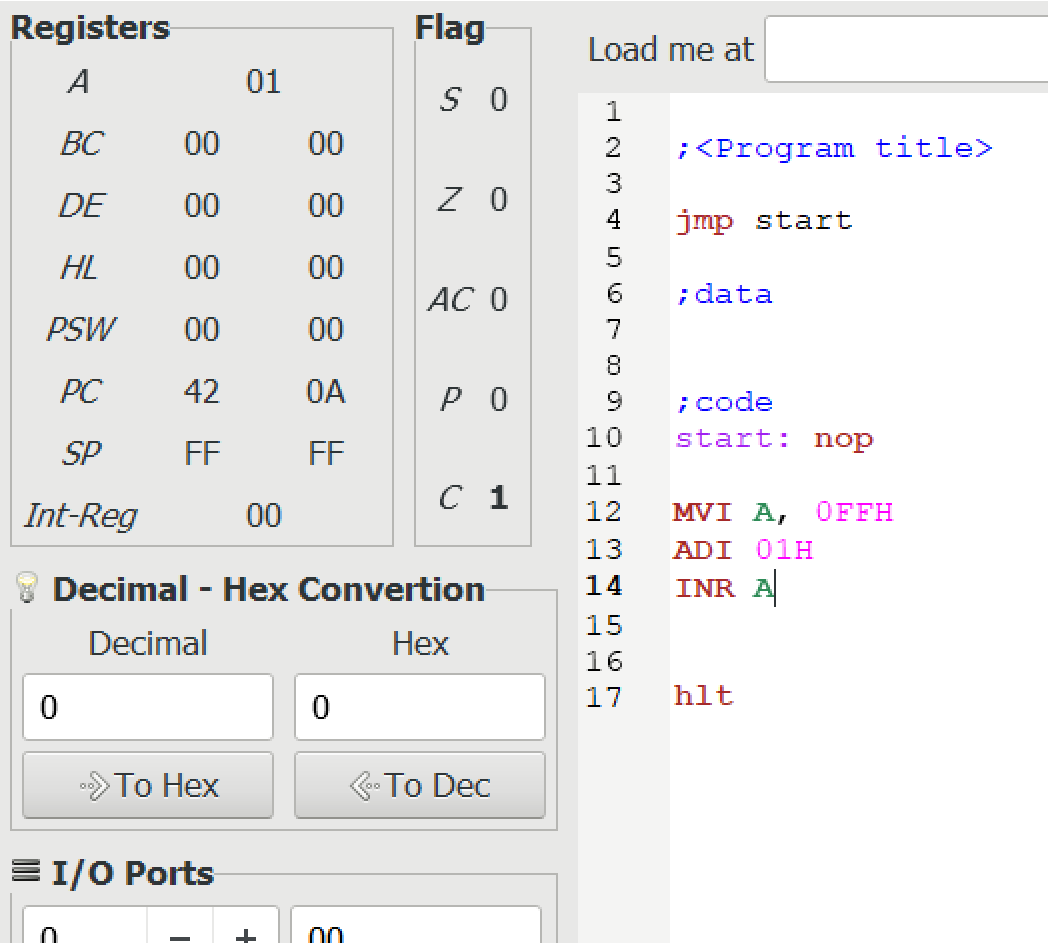
I/O Ports: 0 00

```

;<Program title>

jmp start

;data


;code
start: nop

MVI A, 93H
MVI C, 00B7H
ADD C
ACI 35H
HLT



hlt
```

<u>Question 9</u> : a. Assume the accumulator holds the data byte FFH.By adding 01H comment about the conditions of all flags.

b. After that increment the content of A and comments about the conditions of flags

<u>Output:</u>



Registers

A	01	
BC	00	00
DE	00	00
HL	00	00
PSW	00	00
PC	42	0A
SP	FF	FF
Int-Reg	00	

Flag

S	0
Z	0
AC	0
P	0
C	1

Decimal - Hex Convertion

Decimal	Hex
0	0

```

;<Program title>

jmp start

;data


;code
start: nop

MVI A, 0FFH
ADI 01H
INR A


hlt
```

Question 10 : **Write instruction to do following operations:**

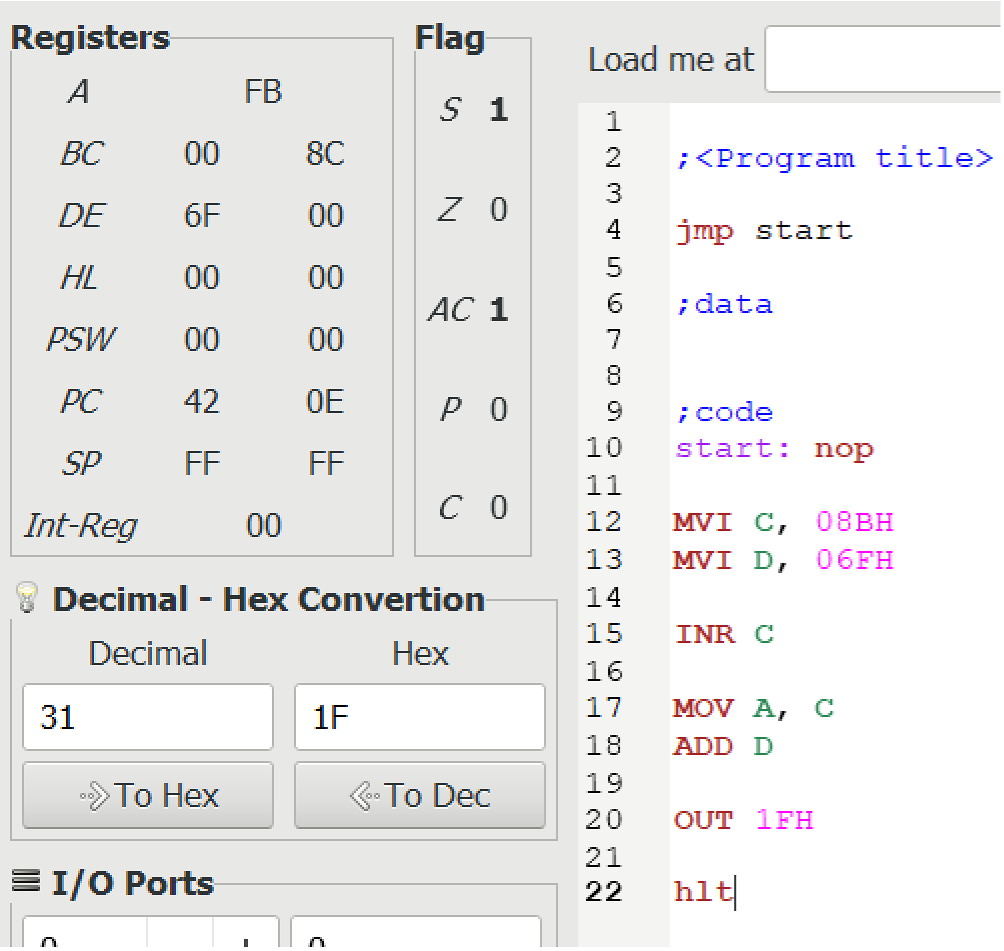
a. Load 8BH in C.

b. Load 6FH in register D.

c. Increment the contents of C by one.

d. Addition of the contents in C and D and display the sum at the PORT address 001FH.

Output:



Registers		
A	FB	
BC	00	8C
DE	6F	00
HL	00	00
PSW	00	00
PC	42	0E
SP	FF	FF
Int-Reg	00	

Flag	
S	1
Z	0
AC	1
P	0
C	0

Decimal	Hex
31	1F

```

;<Program title>

jmp start

;data


;code
start: nop

MVI C, 08BH
MVI D, 06FH

INR C

MOV A, C
ADD D

OUT 1FH

hlt
```

<u>Question 11</u>: Write instruction to do following operations:

a. Load FFH in B.

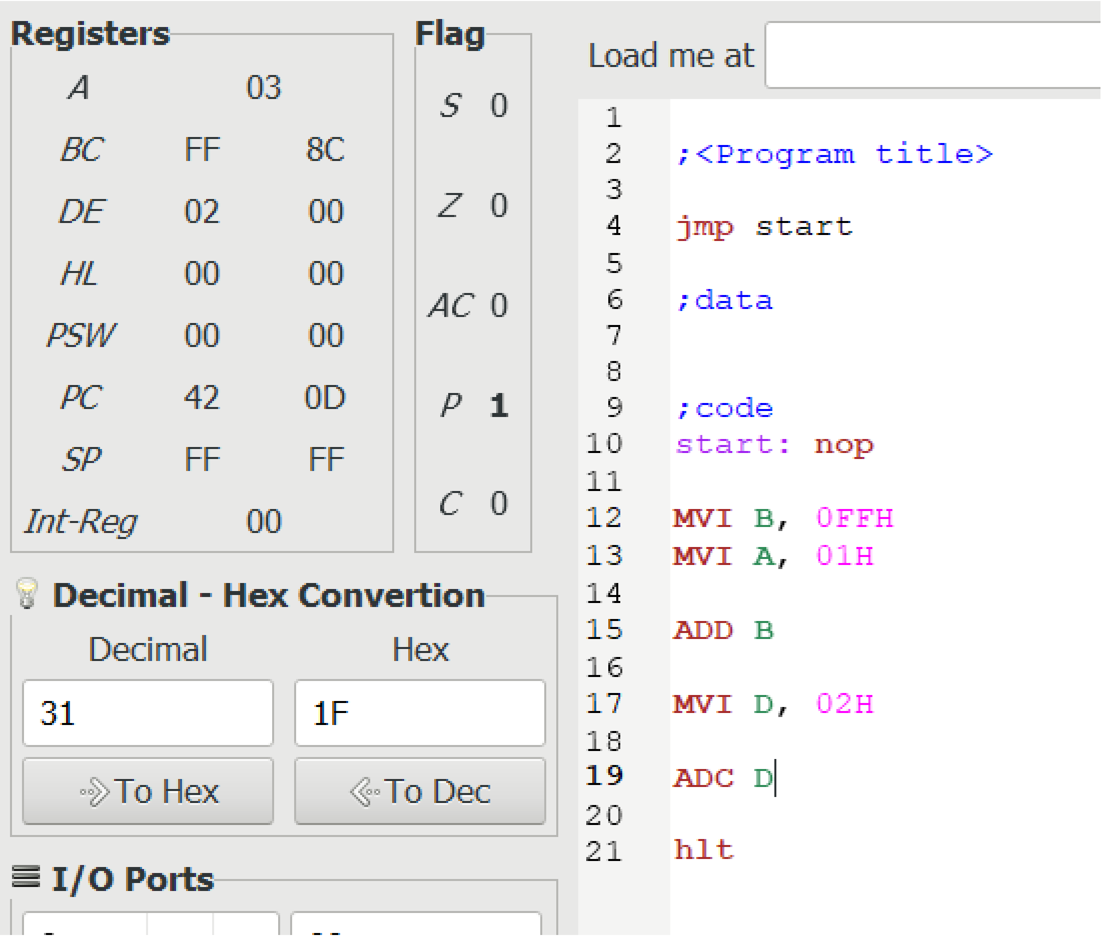
b. Load 01H in A.

c. Add the contents of B with A.

d. Load 02H in register D.

e. Addition of the contents in D with contents of A with carry.

<u>Output:</u>



Registers

A	03	
BC	FF	8C
DE	02	00
HL	00	00
PSW	00	00
PC	42	0D
SP	FF	FF
Int-Reg	00	

Flag

S	0
Z	0
AC	0
P	1
C	0

Decimal - Hex Convertion

Decimal	Hex
31	1F

```
;<Program title>

jmp start

;data


;code
start: nop

MVI B, 0FFH
MVI A, 01H

ADD B

MVI D, 02H

ADC D

hlt
```

Question 12: Write instruction to perform following operations:

a. Perform Logical AND between the contents of register D (D=54H) with the Contents

of Accumulator (A=82H).

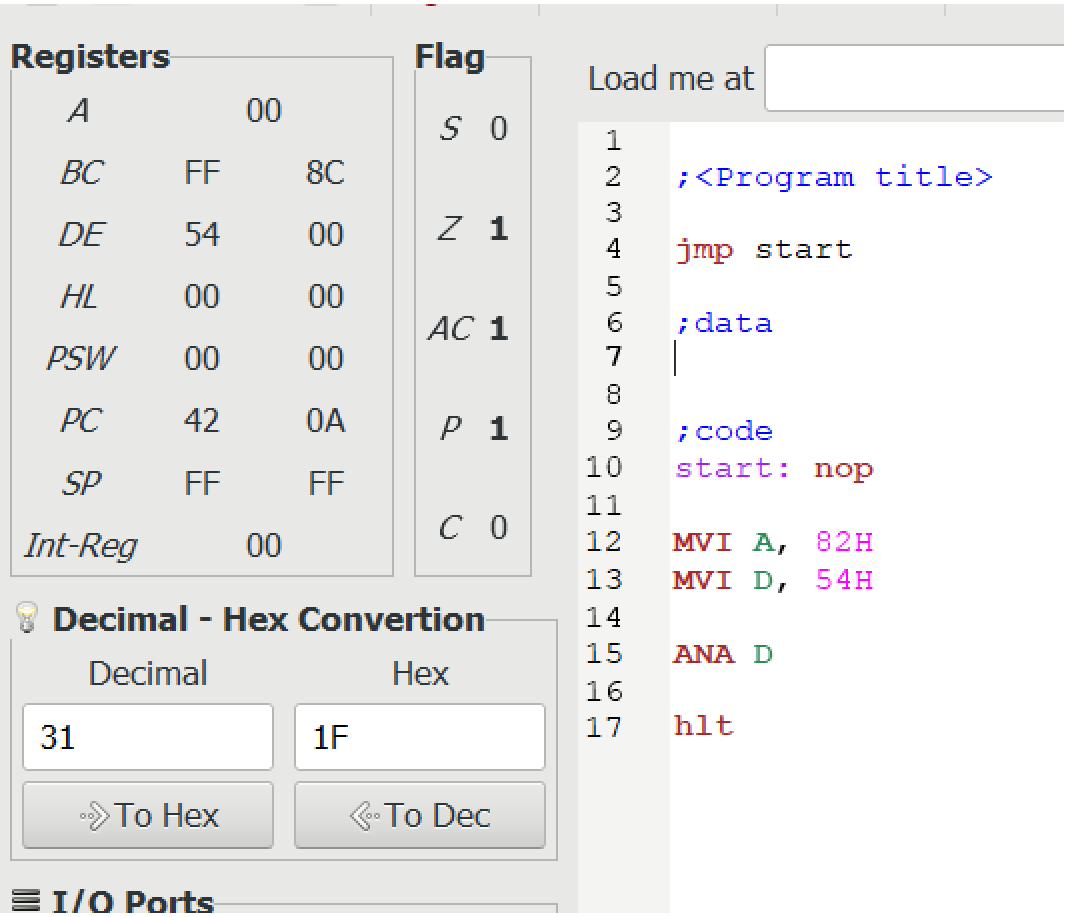
b. Perform Logical AND between the 8-bit data(97H) with the Contents of Accumulator

(A=82H).

c. Perform Logical AND between the contents of memory location with the Contents of

Accumulator (A=72H).

<u>Output:</u>



Registers

A	00	
BC	FF	8C
DE	54	00
HL	00	00
PSW	00	00
PC	42	0A
SP	FF	FF
Int-Reg	00	

Flag

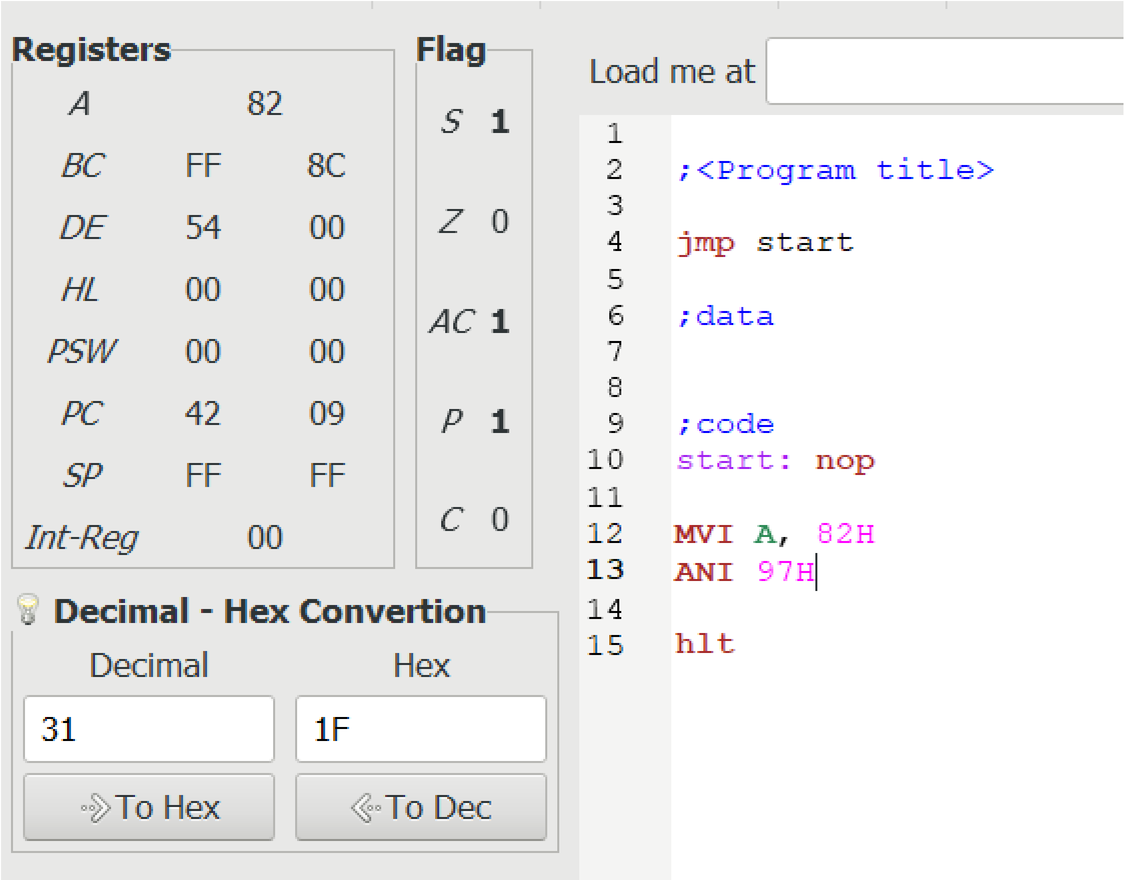
S	0
Z	1
AC	1
P	1
C	0

Decimal - Hex Convertion

Decimal	Hex
31	1F

Load me at

```

;<Program title>

jmp start

;data


;code
start: nop

MVI A, 82H
MVI D, 54H

ANA D

hlt
```



Registers

A	82	
BC	FF	8C
DE	54	00
HL	00	00
PSW	00	00
PC	42	09
SP	FF	FF
Int-Reg	00	

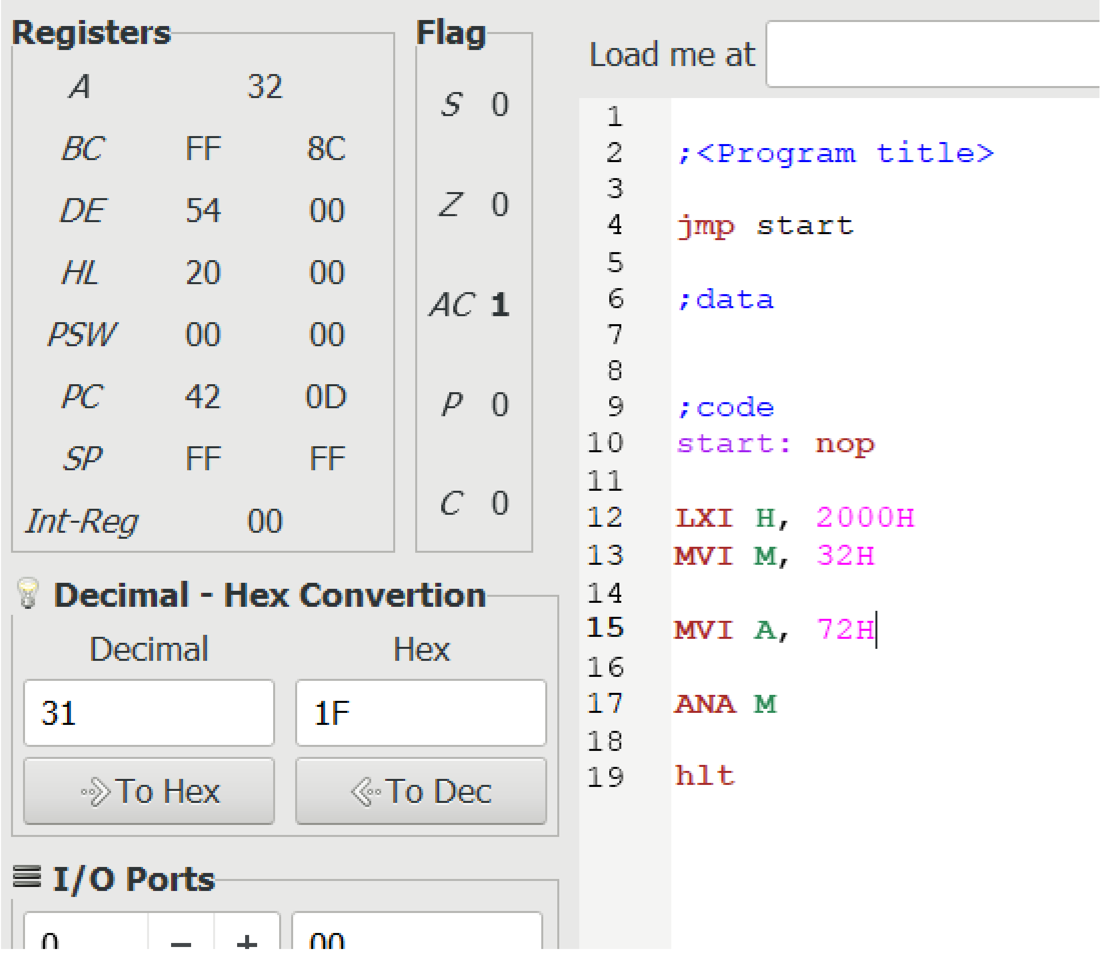
Flag

S	1
Z	0
AC	1
P	1
C	0

Decimal - Hex Convertion

Decimal	Hex
31	1F

```

;<Program title>

jmp start

;data


;code
start: nop

MVI A, 82H
ANI 97H

hlt
```



Registers

A	32	
BC	FF	8C
DE	54	00
HL	20	00
PSW	00	00
PC	42	0D
SP	FF	FF
Int-Reg	00	

Flag

S	0
Z	0
AC	1
P	0
C	0

Decimal - Hex Convertion

Decimal	Hex
31	1F

I/O Ports

0 − + 00

```

;<Program title>

jmp start

;data


;code
start: nop

LXI H, 2000H
MVI M, 32H

MVI A, 72H

ANA M

hlt
```

<u>Question 13</u>: Write instruction to perform following operations:

a. Perform Logical OR between the contents of register B (D=51H) with the Contents of

Accumulator (A=A8H).

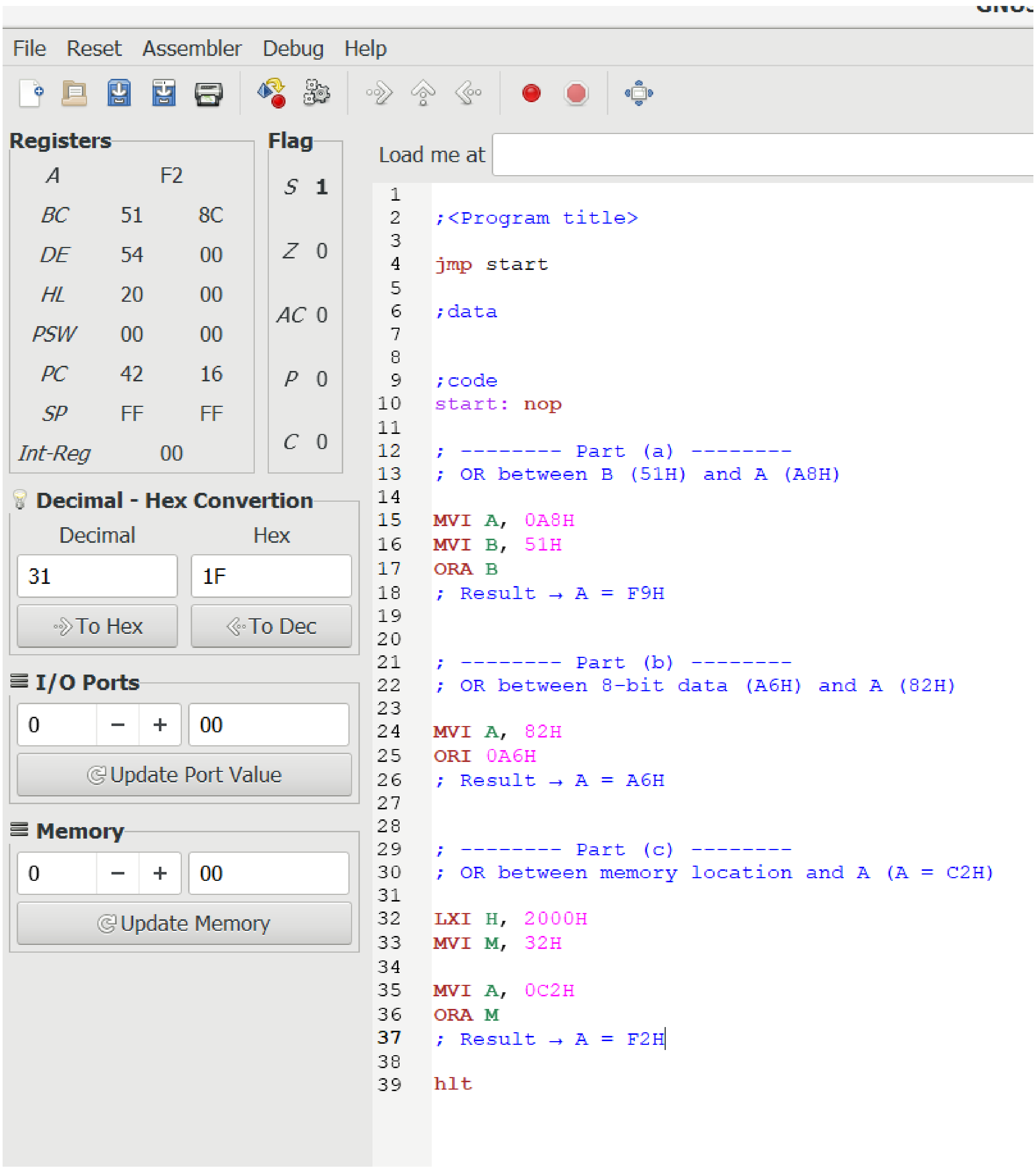
b. Perform Logical OR between the 8-bit data(A6H) with the Contents of Accumulator

(A=82H).

c. Perform Logical OR between the contents of memory location with the Contents of

Accumulator (A=C2H).

<u>Output:</u>



Registers

A	F2	
BC	51	8C
DE	54	00
HL	20	00
PSW	00	00
PC	42	16
SP	FF	FF
Int-Reg	00	

Flag

S	1
Z	0
AC	0
P	0
C	0

Decimal: 31 Hex: 1F

I/O Ports: 0 – + 00

Memory: 0 – + 00

Load me at

```

;<Program title>

jmp start

;data


;code
start: nop

; -------- Part (a) --------
; OR between B (51H) and A (A8H)

MVI A, 0A8H
MVI B, 51H
ORA B
; Result → A = F9H


; -------- Part (b) --------
; OR between 8-bit data (A6H) and A (82H)

MVI A, 82H
ORI 0A6H
; Result → A = A6H


; -------- Part (c) --------
; OR between memory location and A (A = C2H)

LXI H, 2000H
MVI M, 32H

MVI A, 0C2H
ORA M
; Result → A = F2H

hlt
```

<u>Question 14</u>: Write instruction to perform following operations:

a. Perform Logical Exclusive-OR between the contents of register B (D=77H) with the

Contents of Accumulator (A=56H).

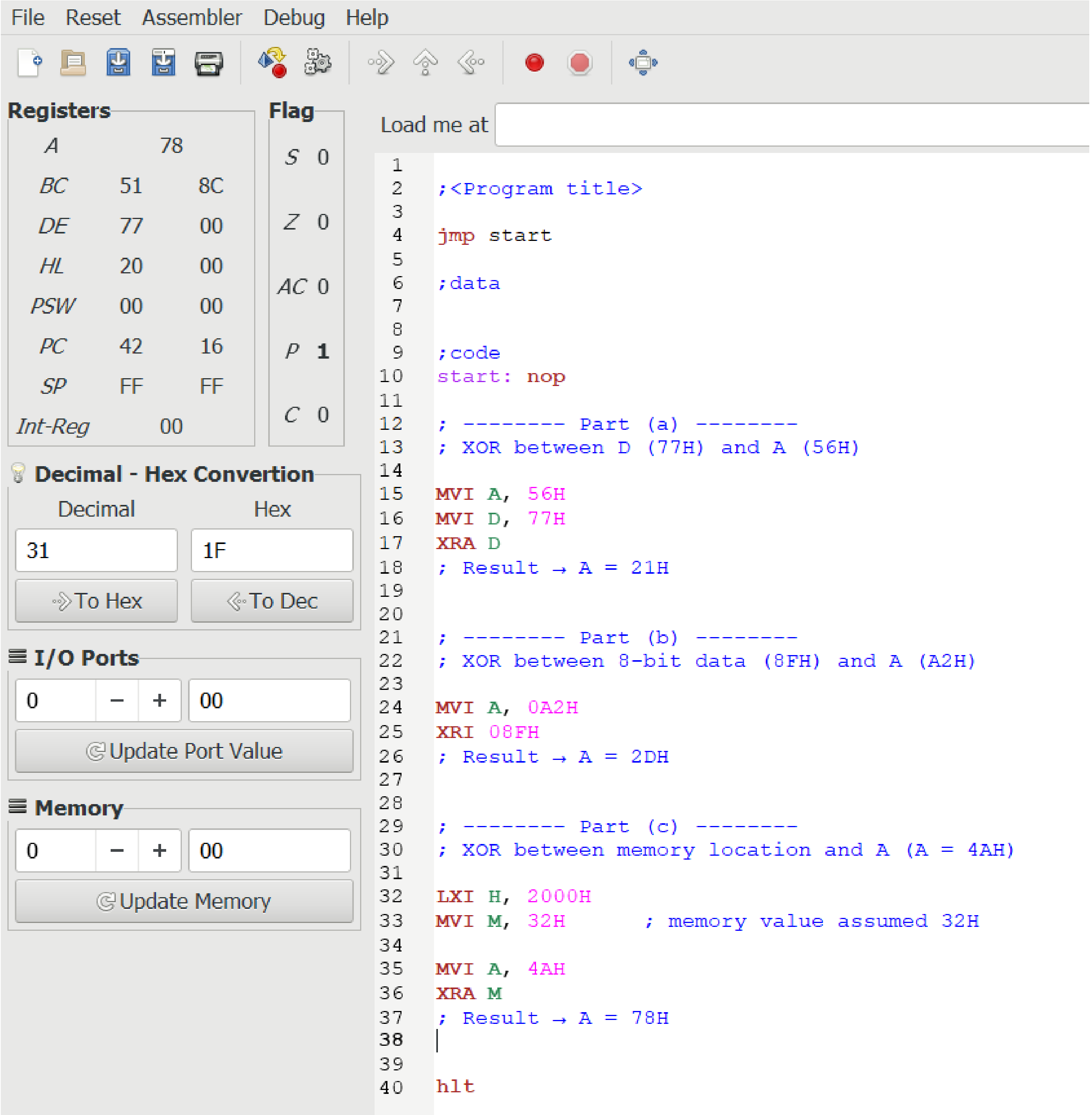
b. Perform Logical Exclusive-OR between the 8-bit data(8FH) with the Contents of

Accumulator (A=A2H).

c. Perform Logical Exclusive-OR between the contents of memory location with the

Contents of Accumulator (A=4AH).

Output:



Registers		
A	78	
BC	51	8C
DE	77	00
HL	20	00
PSW	00	00
PC	42	16
SP	FF	FF
Int-Reg	00	

Flag	
S	0
Z	0
AC	0
P	1
C	0

Decimal: 31 Hex: 1F

I/O Ports: 0 00

Memory: 0 00

```

;<Program title>

jmp start

;data


;code
start: nop

; -------- Part (a) --------
; XOR between D (77H) and A (56H)

MVI A, 56H
MVI D, 77H
XRA D
; Result → A = 21H


; -------- Part (b) --------
; XOR between 8-bit data (8FH) and A (A2H)

MVI A, 0A2H
XRI 08FH
; Result → A = 2DH


; -------- Part (c) --------
; XOR between memory location and A (A = 4AH)

LXI H, 2000H
MVI M, 32H      ; memory value assumed 32H

MVI A, 4AH
XRA M
; Result → A = 78H


hlt
```

<u>Question 15</u>: Write instruction to perform following operations:

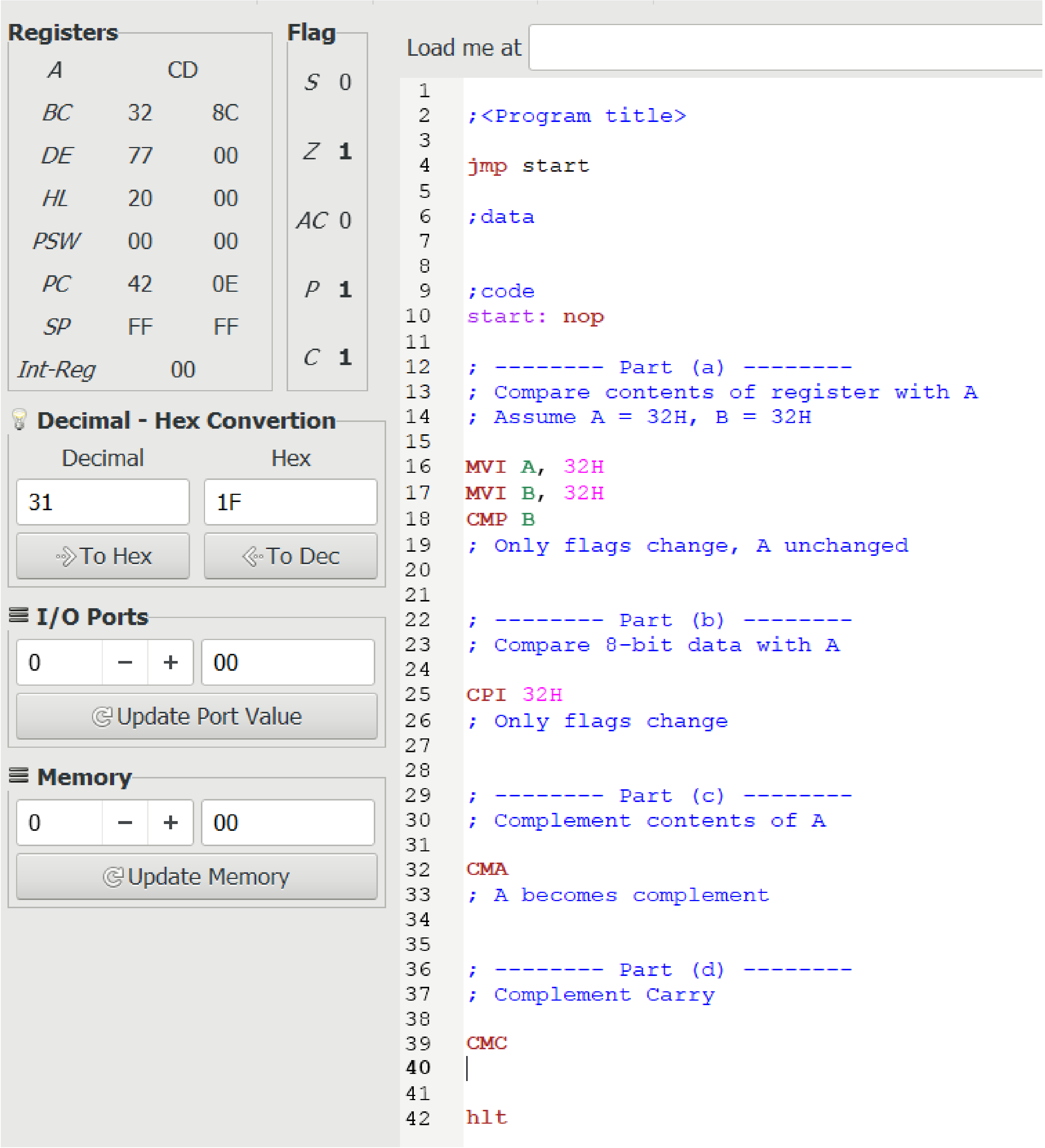
a. Compare the contents of register with the contents of A.

b. Compare the 8-bit data with the contents of A.

c. Complement the contents of A.

d. Complement the Carry

<u>Output:</u>



Registers

A	CD	
BC	32	8C
DE	77	00
HL	20	00
PSW	00	00
PC	42	0E
SP	FF	FF
Int-Reg	00	

Flag

S	0
Z	1
AC	0
P	1
C	1

Decimal - Hex Convertion

Decimal: 31

Hex: 1F

I/O Ports

0 − + 00

Memory

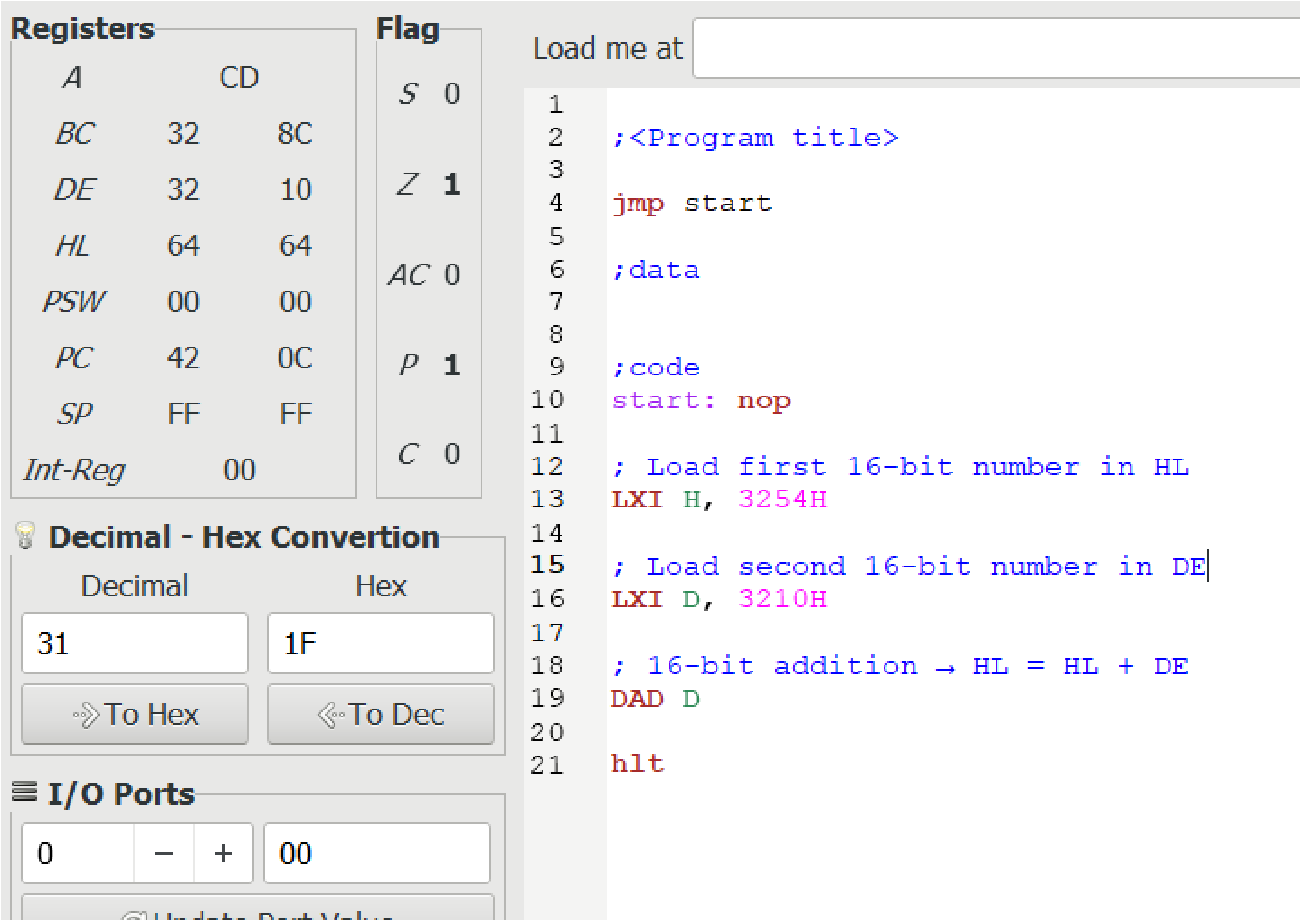
0 − + 00

Load me at

```

;<Program title>

jmp start

;data


;code
start: nop

; -------- Part (a) --------
; Compare contents of register with A
; Assume A = 32H, B = 32H

MVI A, 32H
MVI B, 32H
CMP B
; Only flags change, A unchanged


; -------- Part (b) --------
; Compare 8-bit data with A

CPI 32H
; Only flags change


; -------- Part (c) --------
; Complement contents of A

CMA
; A becomes complement


; -------- Part (d) --------
; Complement Carry

CMC


hlt
```

<u>Question 16</u>: Perform 16-bit Addition

<u>Output:</u>



Registers

A	CD	
BC	32	8C
DE	32	10
HL	64	64
PSW	00	00
PC	42	0C
SP	FF	FF
Int-Reg	00	

Flag

S	0
Z	1
AC	0
P	1
C	0

Decimal	Hex
31	1F

I/O Ports: 0 — 00

```

;<Program title>

jmp start

;data


;code
start: nop

; Load first 16-bit number in HL
LXI H, 3254H

; Load second 16-bit number in DE
LXI D, 3210H

; 16-bit addition → HL = HL + DE
DAD D

hlt
```