Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
M&A Practical 5

Aim : Exercise for jump instruction.

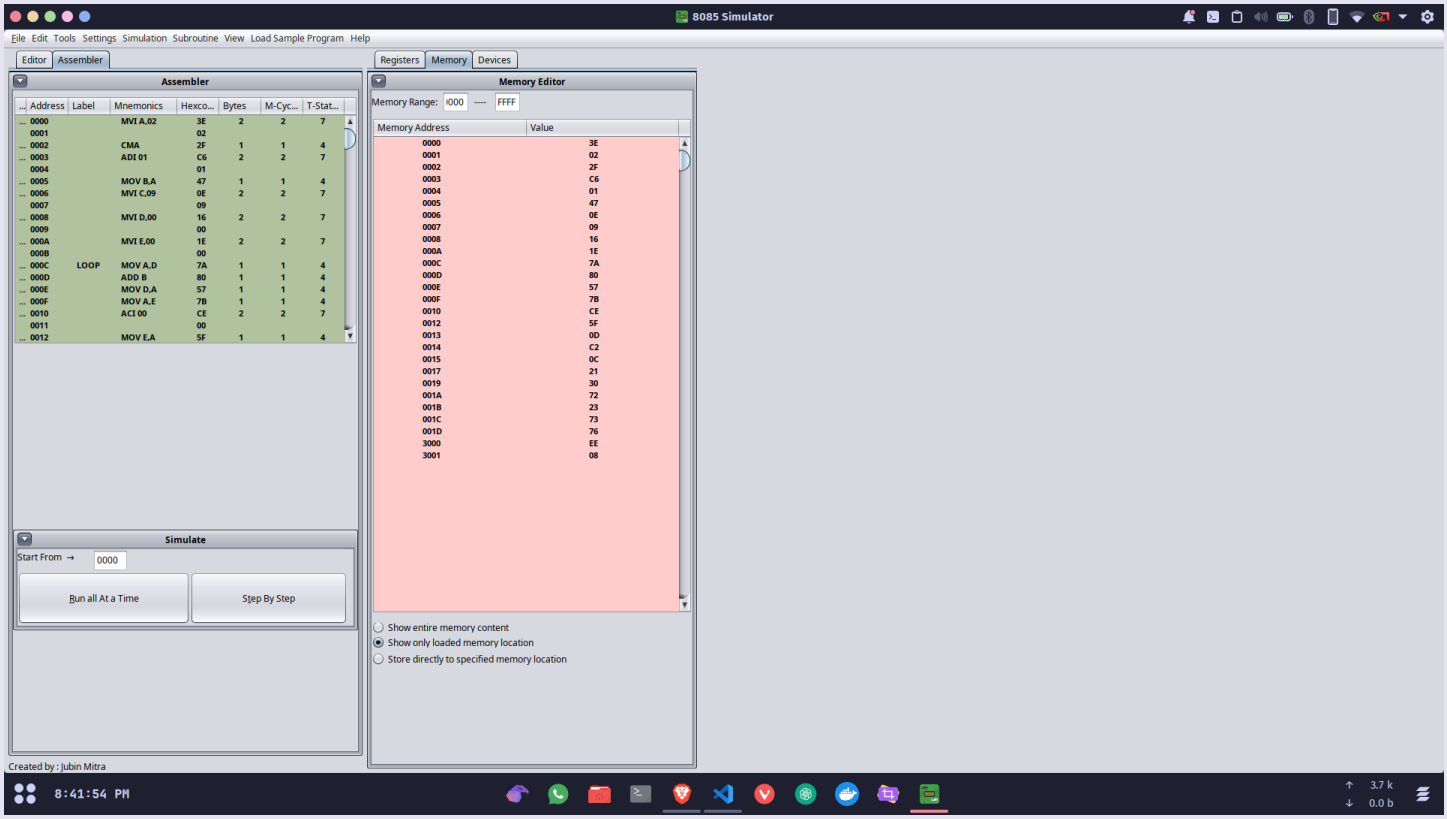
Exercise :

1. **Multiply 2's complement of 02H with 09H and store result into 3000H (LSB) and 3001H (MSB).**

```
MVI A,02H
CMA
ADI 01H

MOV B,A
MVI C,09H
MVI D,00H
MVI E,00H

LOOP:
MOV A,D
ADD B
MOV D,A

MOV A,E
ACI 00H
MOV E,A

DCR C
JNZ LOOP

LXI H,3000H
MOV M,D
INX H
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
M&A Practical 5

```
MOV M,E

HLT
```



Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
M&A Practical 5

PAGE NO.
DATE

Name : Yash Lakhtariya
En no: 21162101012
Branch: CBA Batch-51
subject: M&A Practical 5

```
Code : MVI A,02H  // load 02H in A
       CMA        // 1's complement of A
       ADI 02H    // 2's complement of 02 in A
       MOV B,A    // store A's contents in B
       MVI C,09H  // load 09H in C
       MVI D,00H  // clear D. (For LSB)
       MVI E,00H  // clear E (For MSB)

LOOP :
       MOV A,D    // load from D to A
       ADD B      // add B to A
       MOV D,A    // update D (LSB)
       MOV A,E    // copy from E to A
       ACI 00H    // increment if carry
       MOV E,A    // update E (MSB)
       DCR C      // decrement counter
       JNZ LOOP   // repeat till counter is zero

       LXI H,3000H  // load 3000H in HL pair
       MOV M,D      // store LSB in HL pointer
       INX H        // increment HL
       MOV M,E      // store MSB in HL pointer
       HLT          // end execution
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
M&A Practical 5

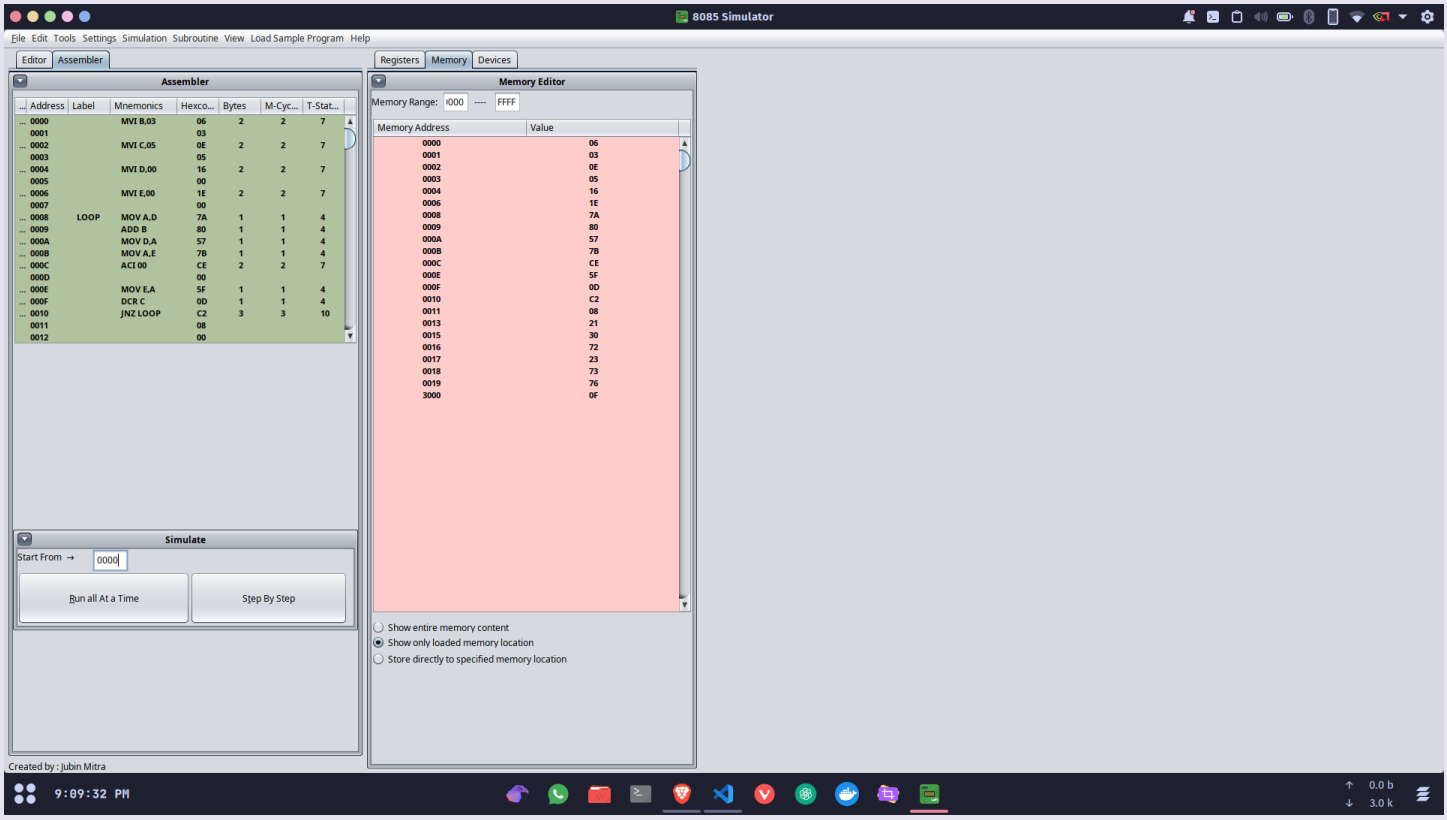
Address	Label	Mnemonics	Hex	Bytes	M-cycles	T-states
0000		MVI A,02	3F	2	2	7
0001			02			
0002		CMA	2F	1	1	4
0003		ADI 01	C6	2	2	7
0004			01			
0005		MOV B,A	47	1	1	4
0006		MVI C,09	0E	2	2	7
0007			09			
0008		MVI D,00	16	2	2	7
0009			00			
000A		MVI E,00	1E	2	2	7
000B			00			
000C	LOOP	MOV A,D	7A	1	1	4
000D		ADD B	80	1	1	4
000E		MOV D,A	57	1	1	4
000F		MOV A,E	7B	1	1	4
0010		ACI 00	CE	2	2	7
0011			00			
0012		MOV E,A	5F	1	1	4
0013		DCR C	0D	1	1	4
0014		JNZ LOOP	C2	3	3	10
0015			0C			
0016			00			
0017		LXI H,3000	21	3	3	10
0018			00			
0019			30			
001A		MOV M,D	72	1	2	7
001B		INX H	23	1	1	6
001C		MOV M,E	73	1	2	7
001D		HLT	76	1	2	5

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
M&A Practical 5

2. **Multiply 03H with 05H and store result into 3000H (LSB) and 3001H (MSB)**

```
MVI B,03H
MVI C,05H
MVI D,00H
MVI E,00H

LOOP:
MOV A,D
ADD B
MOV D,A

MOV A,E
ACI 00H
MOV E,A
DCR C
JNZ LOOP

LXI H,3000H
MOV M,D
INX H
MOV M,E

HLT
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
M&A Practical 5



Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
M&A Practical 5

PAGE NO.
DATE

Code :

```
MVI B,03H    // load 03H in B
MVI C,05H    // load 05H in C
MVI D,00H    // clear D (to store LSB)
MVI E,00H    // clear E (to store MSB)

LOOP:
MOV A,D      // copy from D to A
ADD B        // add B
MOV D,A      // update D (LSB)
MOV A,E      // load MSB (E) to A
ACI 00H      // increment if carry
MOV E,A      // update E (MSB)
DCR C        // decrement counter
JNZ LOOP     // repeat till counter is zero

LXI H,3000H  // load 3000H in HL pair
MOV M,D      // copy from D (LSB) to HL pointer
INX H        // increment HL
MOV M,E      // copy E (MSB) to HL pointer

HLT          // end execution
```

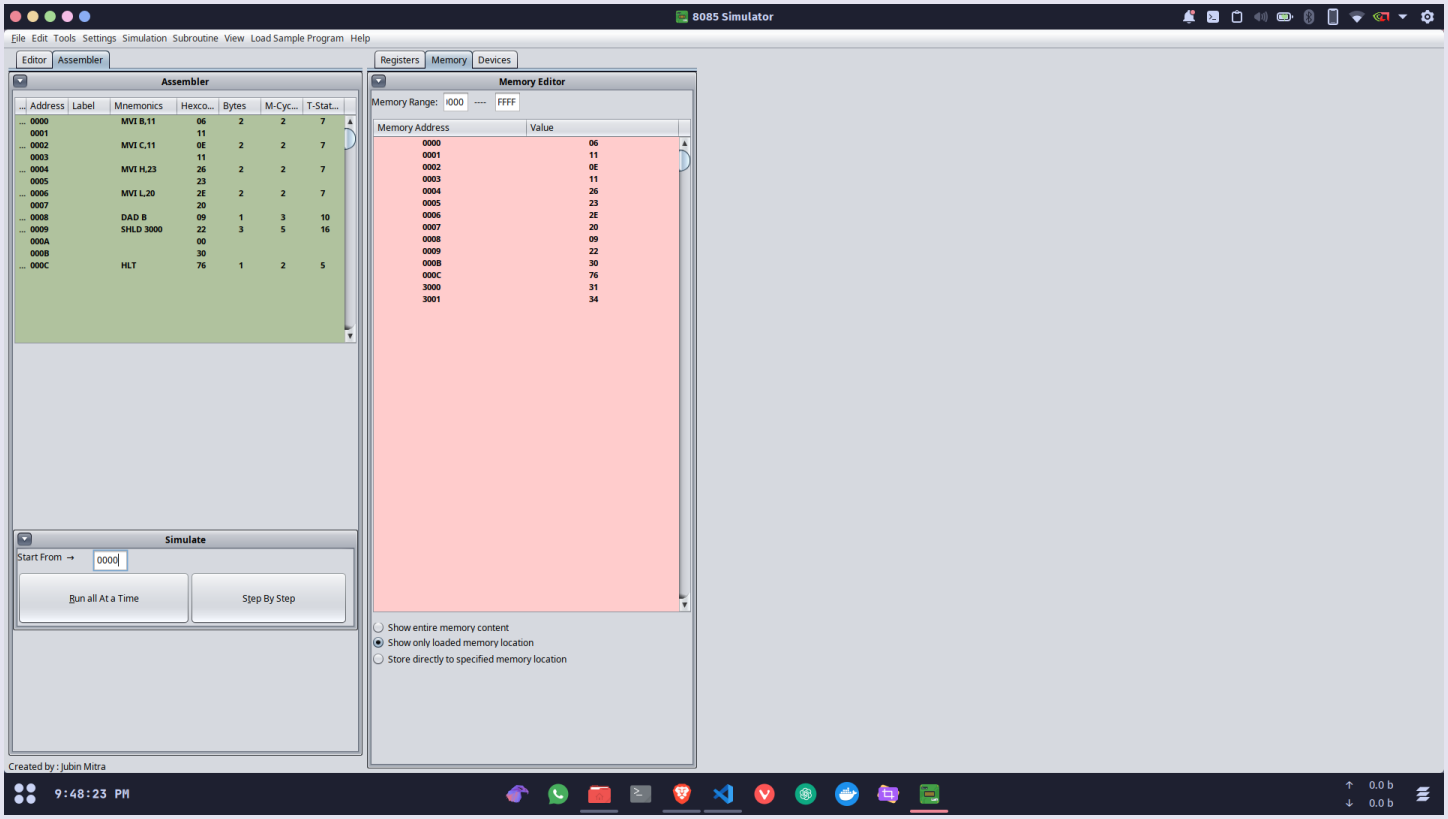
Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
M&A Practical 5

Address	Label	Mnemonics	Hex	Bytes	M-cycles	T-states
0000		MVI B,03	06	2	2	7
0001			03			
0002		MVI C,05	0E	2	2	7
0003			05			
0004		MVI D,00	16	2	2	7
0005			00			
0006		MVI E,00	1E	2	2	7
0007			00			
0008	LOOP	MOV A,D	7A	1	1	4
0009		ADD B	80	1	1	4
000A		MOV D,A	57	1	1	4
000B		MOV A,E	7B	1	1	4
000C		ACI 00	CE	2	2	7
000D			00			
000E		MOV E,A	5F	1	1	4
000F		DCR C	0D	1	1	4
0010		JNZ LOOP	C2	3	3	10
0011			08			
0012			00			
0013		LXI H,3000	21	3	3	10
0014			00			
0015			30			
0016		MOV M,D	72	1	2	7
0017		INX H	23	1	1	6
0018		MOV M,E	73	1	2	7
0019		HLT	76	1	2	5

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
M&A Practical 5

3. **Add two 16 bit numbers with DAD instruction and store it at 3000H with SHLD instruction.**

```
MVI B,11H
MVI C,11H
MVI H,23H
MVI L,20H

DAD B
SHLD 3000H

HLT
```



Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
M&A Practical 5

PAGE NO
DATE

Code:
```
MVI B,11H  // load 11H in B
MVI C,11H  // load 11H in C
MVI H,23H  // load 23H in H
MVI L,20H  // load 20H in L
DAD B      // add
SHLD 3000H
HLT
```

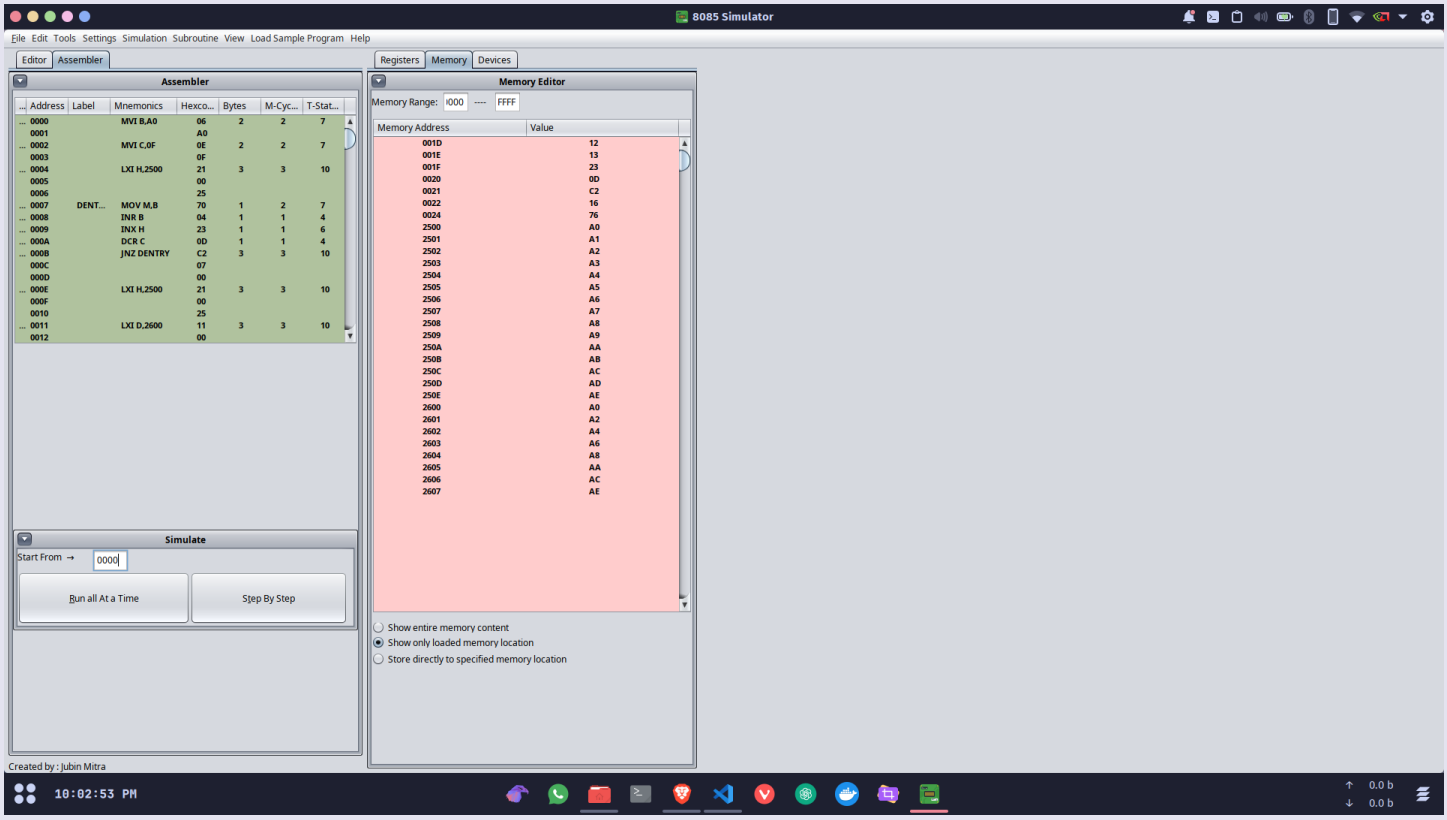
Address	Mnemonics	Hex	Bytes	M-cycles	T-states
0000	MVI B,11	06	2	2	7
0001		11			
0002	MVI C,11	0E	2	2	7
0003		11			
0004	MVI H,23	26	2	2	7
0005		23			
0006	MVI L,20	2E	2	2	7
0007		20			
0008	DAD B	09	1	3	10
0009	SHLD 3000	22	3	5	16
000A		00			
000B		30			
000C	HLT	76	1	2	5

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
M&A Practical 5

4. **Write a program to separate even numbers from the given list of 15 numbers starting from 2500H and store them from 2600H.**

```
        MVI B,A0
        MVI C,0F
        LXI H,2500

DENTRY:     MOV M,B
        INR B
        INX H
        DCR C
        JNZ DENTRY
        LXI H,2500
        LXI D,2600
        MVI C,0F

MAIN:       MOV A,M
        ANI 01
        JNZ ODD
        MOV A,M
        STAX D
        INX D

ODD:    INX H
        DCR C
        JNZ MAIN
        HLT
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
M&A Practical 5



Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
M&A Practical 5

```
Code: MVI B,A0  // load data in B
Code: MVI C,0FH  // load 05H in C (counter)
LXI H,2500H  // load 2500H in HL
DENTRY:
MOV M,B  // copy from B to HL pointer
INR B  // increment data
INX H  // increment addr.
DCR C  // decrement counter
JNZ DENTRY  // repeat till counter is zero

LXI H,2500H  // load addr-1 in HL pair
LXI D,2600H  // load addr-2 in DE pair
MVI C,0FH  // load 0FH [(16)10] in C
MAIN:
MOV A,M  // Copy from HL pointer to A
ANI 01H  // AND A with 01H i.e. non zero
JNZ ODD  // jump if A's contents are odd
MOV A,M  // if even, again load in A from M
STAX D  // store A in DE pointer
INX D  // increment DE pair (addr)
ODD:
INX H  // increment HL pair (addr)
DCR C  // decrement counter
JNZ MAIN  // repeat till counter is zero
HLT  // end execution
```

Address	Mnemonics	Hex	Bytes	M-cycles	T-states
0000	MVI B,A0H	06	2	2	7
0001		A0			
0002	MVI C,0F	0E	2	2	7
0003		0F			
0004	LXI H,2500	21	3	3	10

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
M&A Practical 5

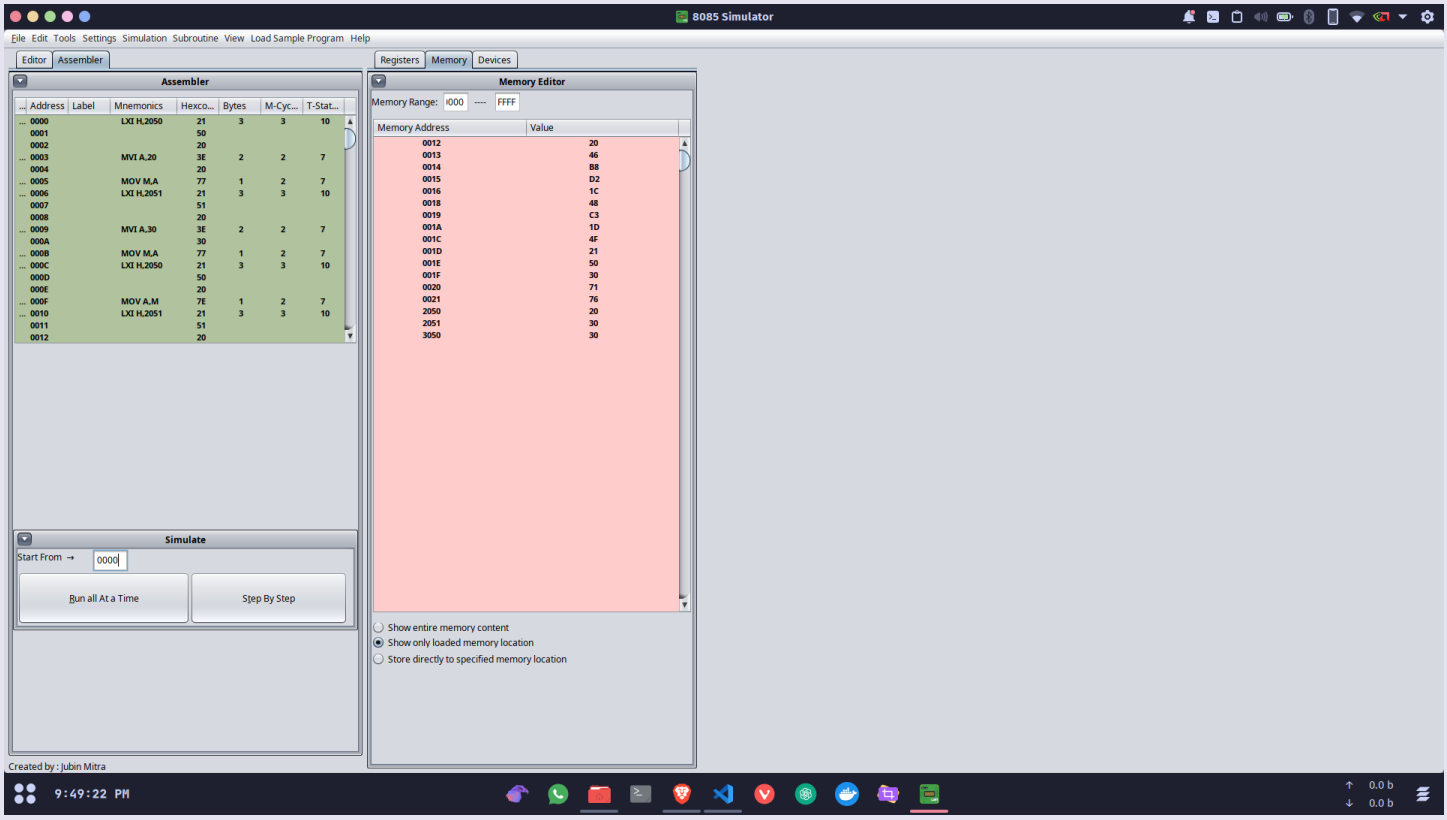
Address	Label	Mnemonics	Hex	Bytes	M-cycles	T-states
0005			00			
0006			25			
0007	PENTRY	MOV M, B	70	1	2	7
0008		INR B	04	1	1	4
0009		INX H	23	1	1	6
000A		DCR C	0D	1	1	4
000B		JNZ PENTRY	C2	3	3	10
000C			07			
000D			00			
000E		LXI H, 2500	21	3	3	10
000F			00			
0010			25			
0011		LXI D, 2600	11	3	3	10
0012			00			
0013			26			
0014		MVI C, 0F	0E	2	2	7
0015			0F			
0016	MAIN	MOV A, M	7E	1	2	7
0017		ANI 01	E6	2	2	7
0018			01			
0019		JNZ ODD	C2	3	3	10
001A			1F			
001B			00			
001C		MOV A, M	7E	1	2	7
001D		STAX D	12	1	2	7
001E		INX D	13	1	1	6
001F	ODD	INX H	23	1	1	6
0020		DCR C	0D	1	1	4
0021		JNZ MAIN	C2	3	3	10
0022			16			
0023			00			
0024		HLT	76	1	2	5

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
M&A Practical 5

5. **Write a program to find a maximum of two given 8-bit number stored at 2050H and 2051H. Store the largest value at 3050H.**

```
LXI H,2050H
MVI A,20H
MOV M,A

LXI H,2051H
MVI A,30H
MOV M,A

//---------------------

LXI H,2050H
MOV A,M
LXI H,2051H
MOV B,M

CMP B

JNC ABIG
MOV C,B
JMP EXITLOOP
ABIG:
MOV C,A

EXITLOOP:
LXI H,3050H
MOV M,C

HLT
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
M&A Practical 5



Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
M&A Practical 5

Code :

```
LXI H,2050H     // load 2050H in HL pair
MVI A,20H       // load 20H in A
MOV M,A         // store A in HL pointer
LXI H,2051H     // load 2051H in HL pair
MVI A,30H       // load A with 30H
MOV M,A         // copy A in HL pointer

LXI H,2050H     // load 2050H in HL pair
MOV A,M         // copy from HL pointer to A
LXI H,2051H     // load 2051H in HL pair
MOV B,M         // copy from HL pointer to B
CMP B           // compare A with B
JNC ABIG        // jump if not carry
MOV C,B         // copy from B to C (answer)
JMP EXITLOOP    // unconditionally jump
ABIG:
MOV C,A         // if A>B, load A in C
EXIT LOOP:
LXI H,3050H     // load 3050H in HL pair
MOV M,C         // copy from C to HL pointer
HLT             // end execution
```

Address	Mnemonics	Hex	Bytes	M-cycles	T-states
0000	LXI H,2050	21	3	3	10
0001		50			
0002		20			
0003	MVI A,20	3E	2	2	7
0004		20			
0005	MOV M,A	77	1	2	7
0006	LXI H,2051	21	3	3	10
0007		51			

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
M&A Practical 5

PAGE NO.
DATE

Address	Label	Mnemonics	Hex	Bytes	M-cycles	T-states
0008			20			
0009		MVI A, 30	3E	2	2	7
000A			30			
000B		MOV M, A	77	1	2	7
000C		LXI H, 2050	21	3	3	10
000D			50			
000E			20			
000F		MOV A, M	7F	1	2	7
0010		LXI H, 2051	21	3	3	10
0011			51			
0012			20			
0013		MOV B, M	46	1	2	7
0014		CMP B	B8	1	1	4
0015		JNC ABIG	D2	3	3	10
0016			1C			
0017			00			
0018		MOV C, B	48	1	1	4
0019		JMP EXITLOOP	C3	3	3	10
001A			1D			
001B			00			
001C	ABIG	MOV C, A	4F	1	1	4
001D	EXITLOOP	LXI H, 3050	21	3	3	10
001E			50			
001F			30			
0020		MOV M, C	71	1	2	7
0021		HLT	76	1	2	5