**Practical list of 8085 simulator**

1. **Addition of two 8 bit numbers having 16 bit sum.**

***Algo:***

1. Start the program.

2. Load the first data in the accumulator.

3. Initialize the carry with zero.

4.Move value at memory to the accumulator.

5.Now load value of second data.

6.Add content at Memory and accumulator and store it in Accumulator.

7.If the carry is “0” then store the result which is specified.

8.If the carry is “1” then increment the C register and store the result.

9.Stop the program.

***Program:***

// ADDITION OF TWO 8 BIT NUMBERS HAVING 16 BIT SUM

// Manually strore 1st no in the memory location C050

// Manually store 2nd no in the memory location C051

// Result is stored in C052, & C053

# BEGIN 0000H

LXI H,C050

MVI C,00

MOV A,M

INX H

ADD M

JNC AHEAD

INR C

AHEAD: STA C053

MOV A,C

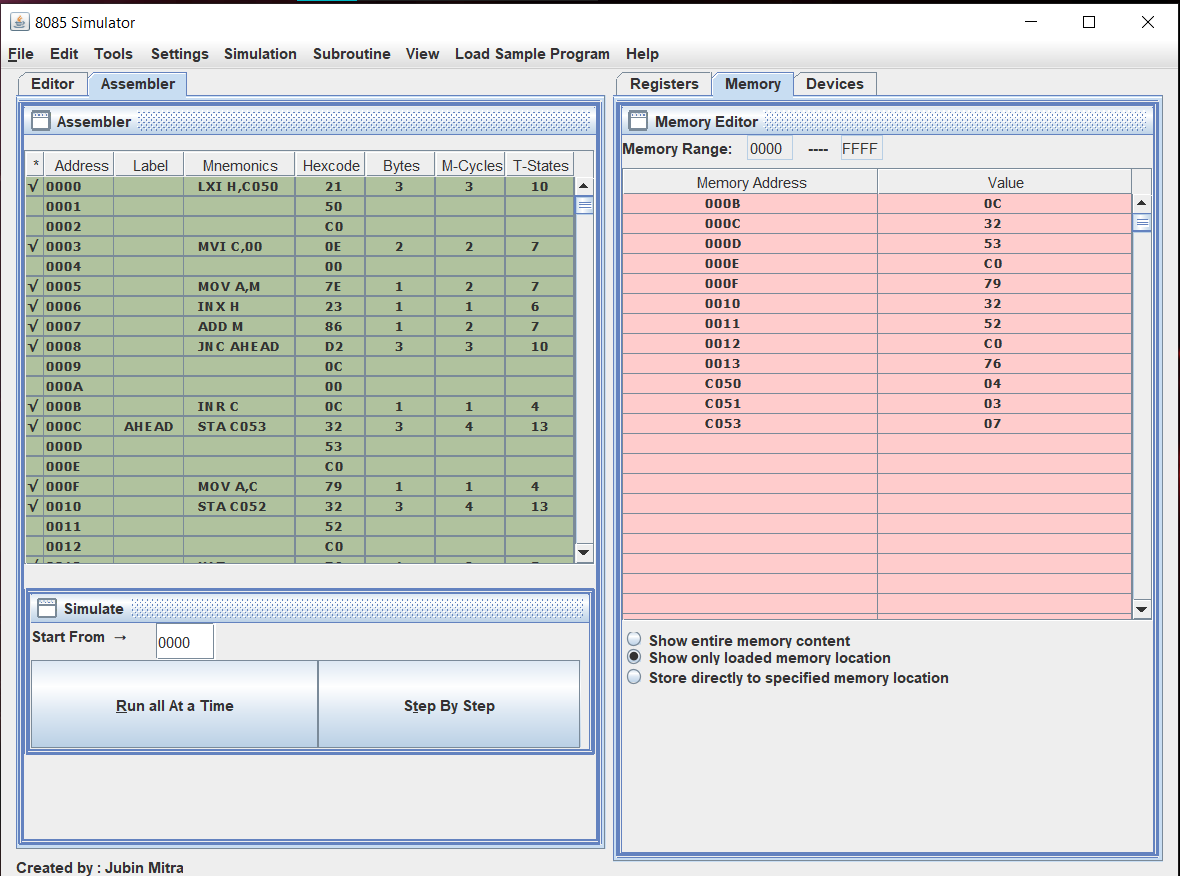
STA C052

HLT

# ORG C050

# DB 04H,03H

**Output:**

****

1. **Subtraction of two 8 bit numbers (Single program should satisfy the following cases)**

**Case:1 When minuend is greater than subtrahend**

**Case 2: When minuend is smaller than subtrahend.**

***Algo:***

1. Start the program.

2. Load the first data in the accumulator.

3. Initialize the carry with zero.

4.Move value at memory to the accumulator.

5.Now load value of second data.

6.Sub content at Memory and accumulator(M-A) and store it in Accumulator.

7.If the carry is “0” then store the result which is specified.

8.If the carry is “1” then increment the C register and store the result.

9.Stop the program.

***Program:***

//8 BIT SUBSTRACTION

# BEGIN 0000H

LXI H,C050

MVI C,00

MOV A,M

INX H

SUB M

JNC AHEAD

INR C

AHEAD: STA C053

MOV A,C

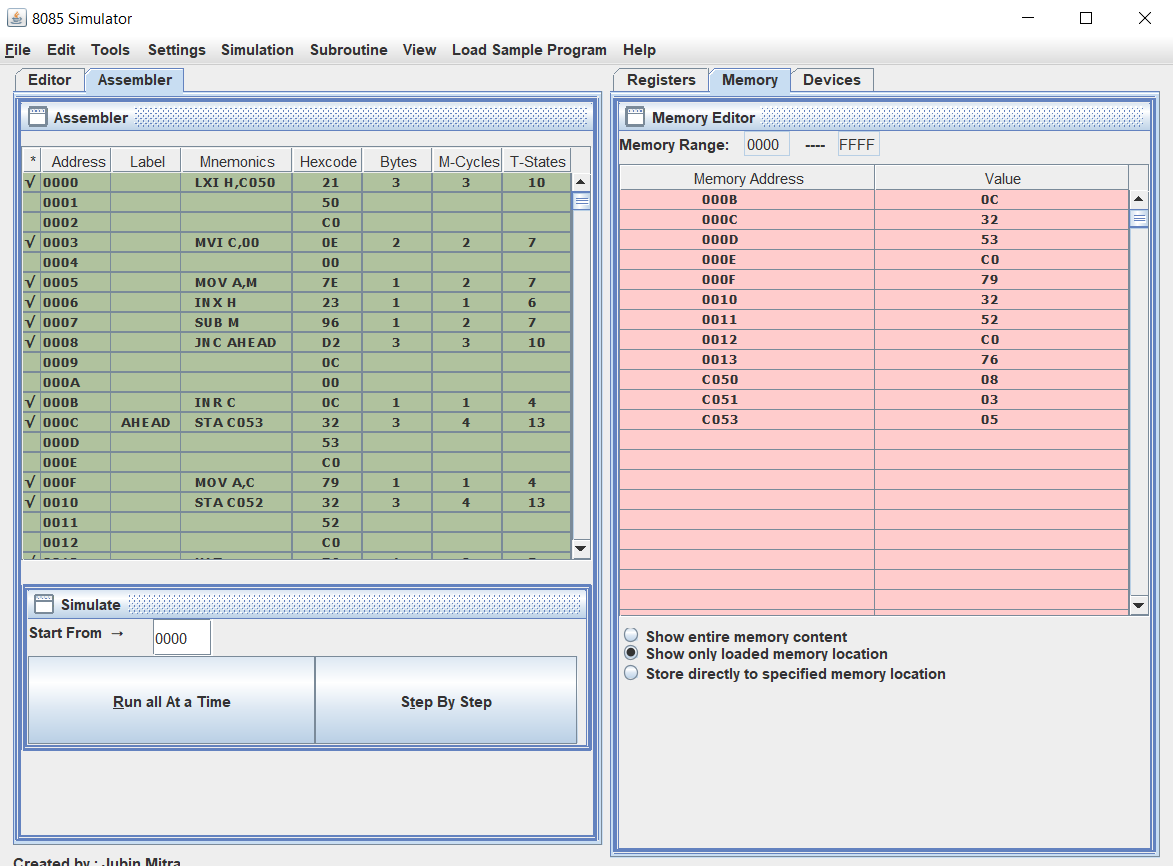
STA C052

HLT

# ORG C050

# DB 08H,03H

**Output:**

****

1. **Multiplication two 8 bit numbers, result is 16 bit number.**

***Algo:***

1. Start the program.

2. Load the first data in the accumulator.

3.Move value at Accumulator to C.

4.Load the second data in the accumulator.

5.Move value at Accumulator to B.

6.Now clear accumulator by moving 00 to accumulator.

7.Create a loop in which the second data is added with itself to get multiplication of two numbers upto no. of second data. **i.e.** if the number is 5 and 2 then we have to add 2 to 2 for 5 times ,2+2+2+2+2=10.

8.Store the data.

9.Stop the program.

***Program:***

# BEGIN 0000H

LDA C050H

MOV C,A

LDA C051H

MOV B,A

MVI A,00H

LOOP: ADD B

DCR C

JNZ LOOP

STA C052H

MVI A,00H

JNC LABEL

INR A

LABEL: STA C053H

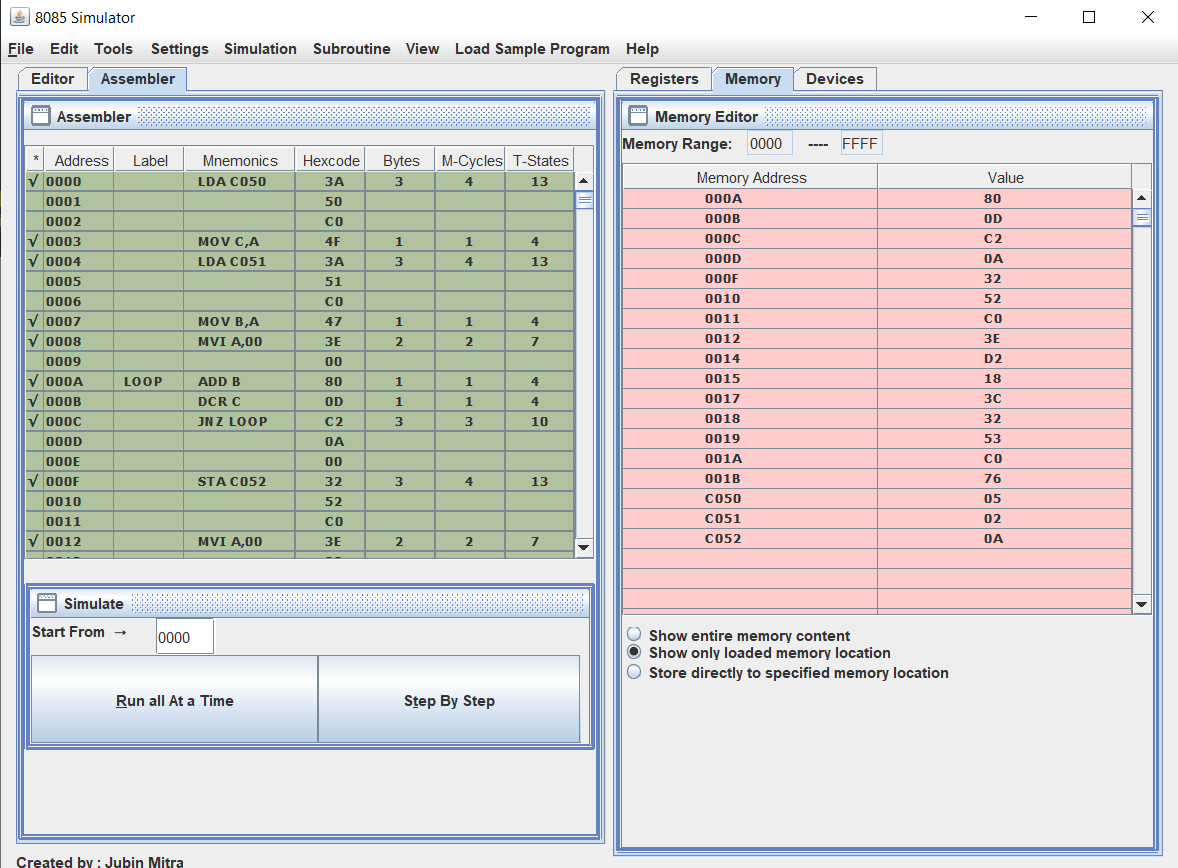
HLT

// multiplication of two numbers

# ORG C050

# DB 05H,02H

**Output:**



1. **Division of two 8 bit number.**

***Algo:***

1.Start the Program.

2.Load value of divisior.

3.Move value of Accumulator to B.

4.Load value of dividend.

5. Initialize the carry with zero.

6. Compare the 8 bit instructions in A and B registers. Subtract B register from accumulator & increment the value in C register.

7.Increment the HL pair and move the content in accumulator to memory.

8.Store the value at specific memory location.

9.Stop the program.

***Program:***

# ORG 2000H

# BEGIN 2000H

LDA 2502 // divisor

MOV B,A

LDA 2501 //dividend

MVI C,00

LOOP: CMP B

JC AHEAD

SUB B

INR C

JMP LOOP

AHEAD:

STA 2503 //Remainder

MOV A,C

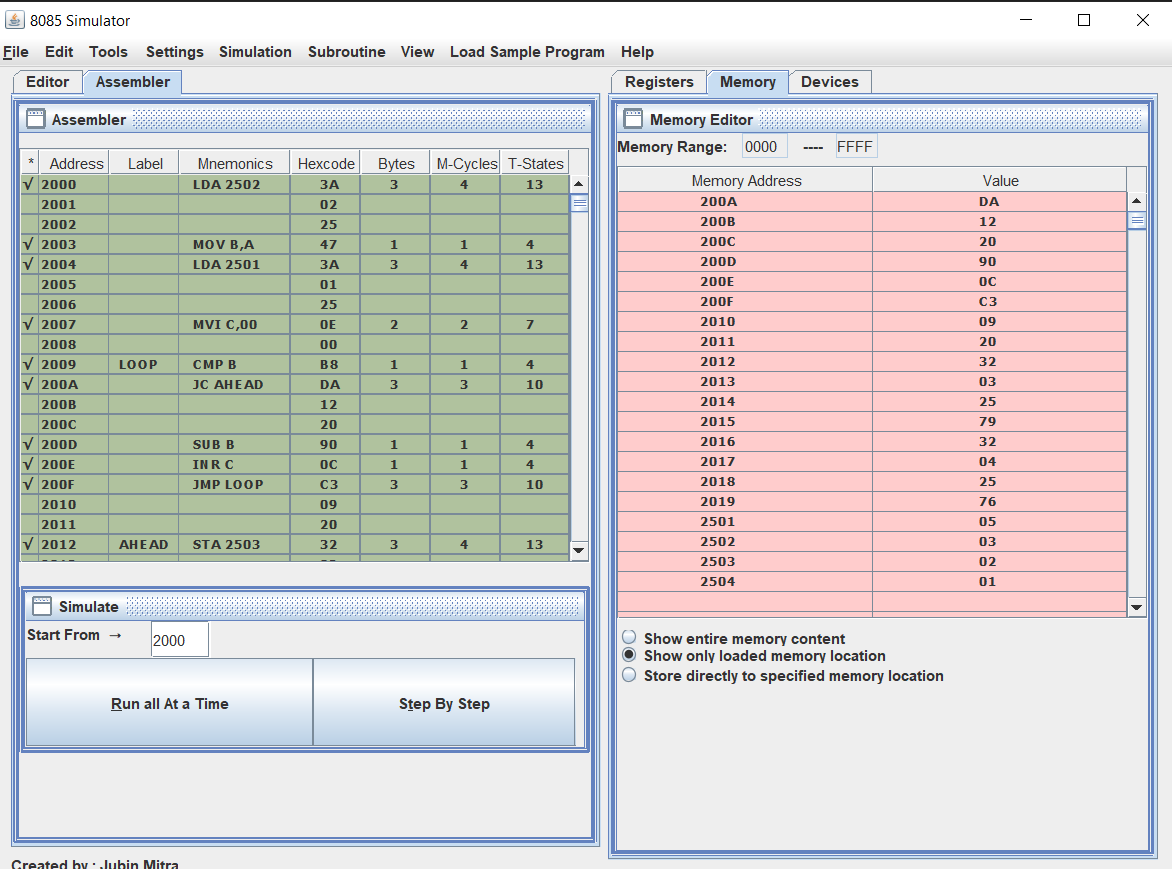
STA 2504 //Quotient

HLT

# ORG 2501H

# DB 05H,03H

**Output:**



1. **Write a 8085 program to find largest number in the given array of numbers.**

***Algo:***

1. Place the elements of an array in consecutive memory locations.

2. Fetch the first element from the array.

3. Initialize a counter (register) with the total number of elements in an array.

4. Decrement the counter by 1.

5. Increment the memory pointer to point to the next element.

6. Compare the accumulator content with the memory content (next element).

7. If the accumulator content is smaller, then move the memory content (largest element) to the accumulator. Else continue.

8. Decrement the counter by 1.

9. Repeat steps 5 to 8 until the counter reaches zero

10. Store the result (accumulator content) in the specified memory location.

***Program:***

// Highest number in an array

# BEGIN 0000H

LXI H,C050

MVI C,0A

MOV A,M

LABEL: CMP M

JNC LOOP

MOV A,M

LOOP: INX H

DCR C

JZ LABEL1

JMP LABEL

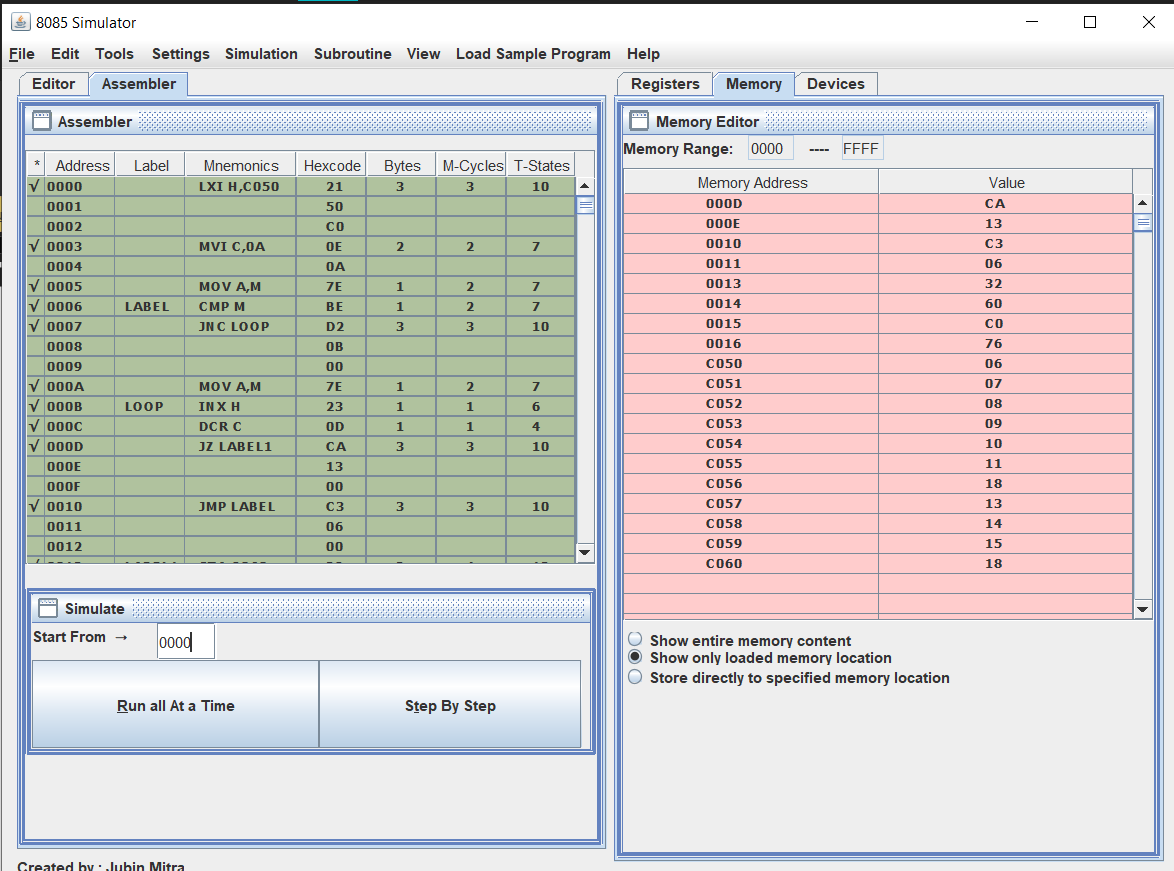
LABEL1: STA C060

HLT

# ORG C050H

# DB 06H,07H,08H,09H,10H,11H,18H,13H,14H,15H

**Output:**

****

1. **Write a 8085 program to find smallest number in the given array of numbers.**

***Algo:***

1. Place the elements of an array in consecutive memory locations.

2. Fetch the first element from the array.

3. Initialize a counter (register) with the total number of elements in an array.

4. Decrement the counter by 1.

5. Increment the memory pointer to point to the next element.

6. Compare the accumulator content with the memory content (next element).

7. If the accumulator content is smaller, then move the memory content (smallest element) to the accumulator. Else continue.

8. Decrement the counter by 1.

9. Repeat steps 5 to 8 until the counter reaches zero

10. Store the result (accumulator content) in the specified memory location.

***Program:***

// Smallest number in an array

# BEGIN 0000H

LXI H,C050

MVI C,0A

MOV A,M

LABEL: CMP M

JC LOOP

MOV A,M

LOOP: INX H

DCR C

JZ LABEL1

JMP LABEL

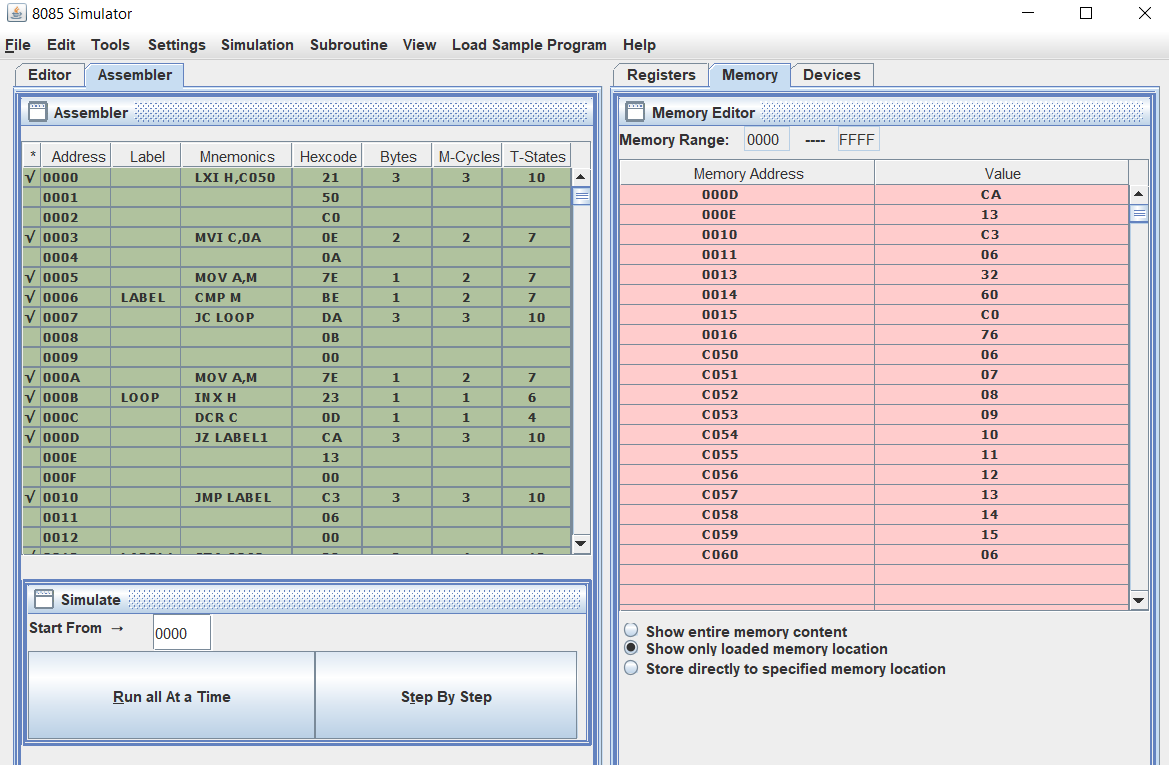
LABEL1: STA C060

HLT

# ORG C050H

# DB 06H,07H,08H,09H,10H,11H,12H,13H,14H,15H

**Output:**



1. **Write a 8085 program to sort the given array of numbers in ascending order.**

***Algo:***

1. Get the numbers to be sorted from the memory locations.

2.Compare the first two number and if the first number is larger than second than interchange the number.

3.If the first number is smaller go to step 4.

4.Repeat the step 2 and 3 until the numbers are in required order.

***Program:***

LXI H,C050

MOV C,M

DCR C

REPEAT: MOV D,C

LXI H,C051

LOOP: MOV A,M

INX H

CMP M

JC SKIP

MOV B,M

MOV M,A

DCX H

MOV M,B

INX H

SKIP: DCR D

JNZ LOOP

DCR C

JNZ REPEAT

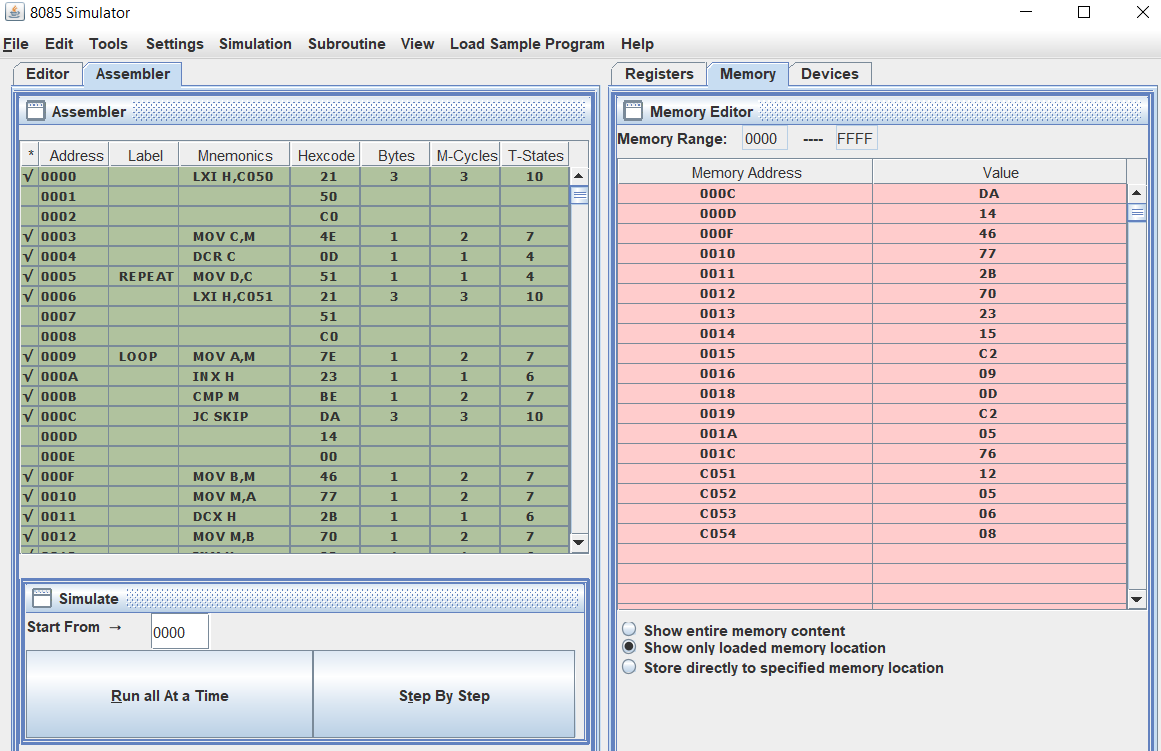
HLT

# ORG C051H

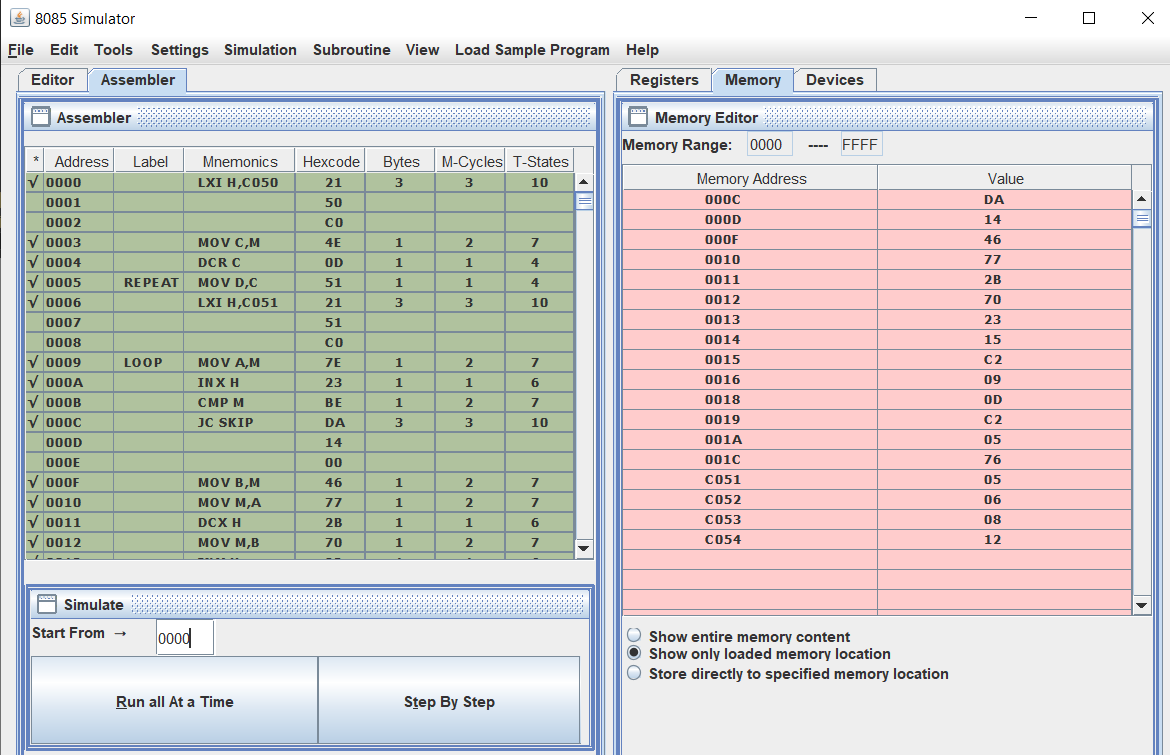
# DB 12H,05H,06H,08H

**Output:**

**Before**

****

**After:**

****

1. **Write a 8085 program to sort the given array of numbers in descending order.**

***Algo:***

1. Get the numbers to be sorted from the memory locations.

2.Compare the first two number and if the first number is smaller than second than interchange the number.

3.If the first number is smaller go to step 4.

4.Repeat the step 2 and 3 until the numbers are in required order.

***Program:***

# BEGIN 0000H

LXI H,C050

MOV C,M

DCR C

REPEAT: MOV D,C

LXI H,C051

LOOP: MOV A,M

INX H

CMP M

JNC SKIP

MOV B,M

MOV M,A

DCX H

MOV M,B

INX H

SKIP: DCR D

JNZ LOOP

DCR C

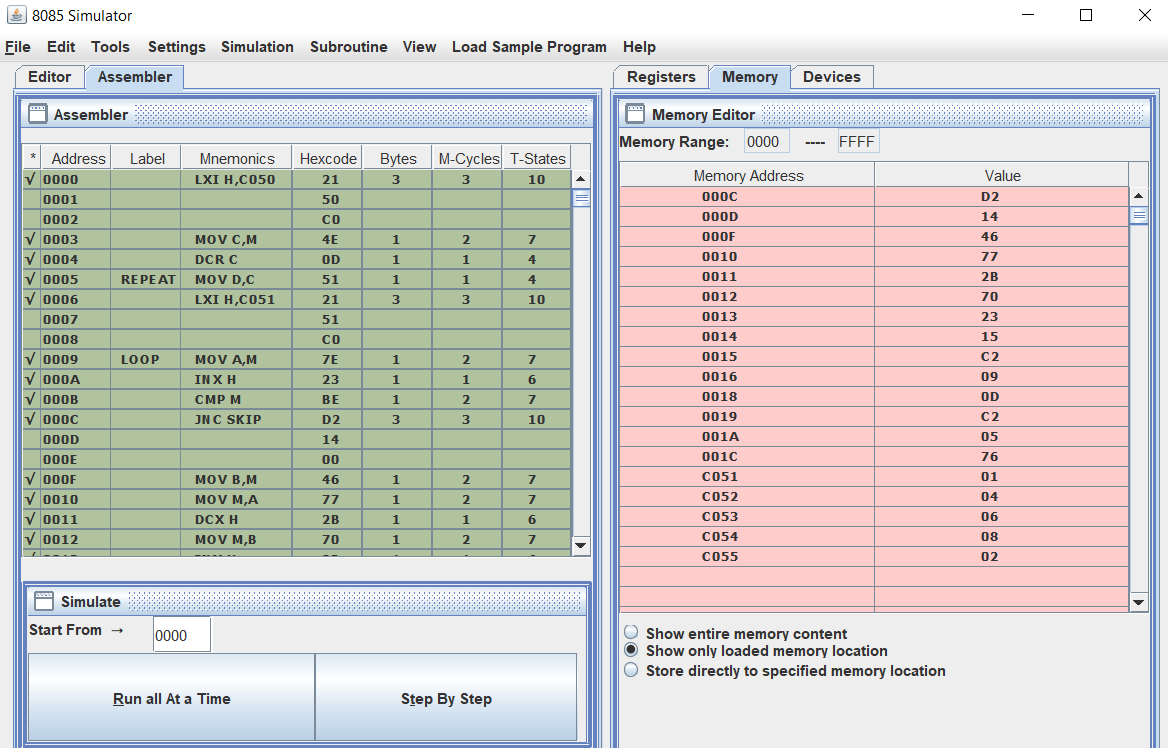
JNZ REPEAT

HLT

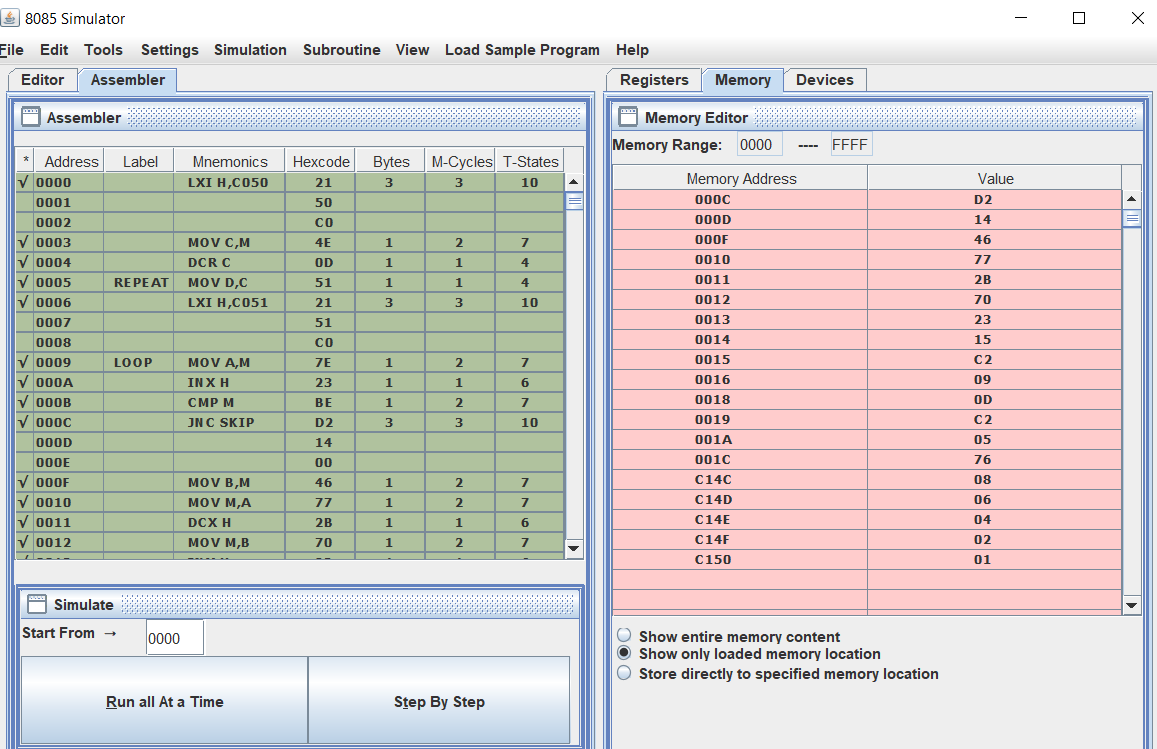
# ORG C051H

# DB 01H,04H,06H,08H,02H

**Output:**

****

**After:**

****

1. **Write a 8085 program to convert a given hexadecimal number to BCD number.**

***Algo:***

1)Initialize memory pointer to 4150H location.

2) Get the Hexadecimal number in C-register.

3)Perform repeated addition for C number of times.

4) Adjust for BCD at each step.

5) Store the BCD data in Memory.

***Program:***

LXI H,C050 // Initialize memory pointer

MVI D,00 // Clear D- reg for Most significant Byte

XRA A // Clear Accumulator

MOV C,M // Get HEX data

LOOP2: ADI 01 // Count the number one by one

DAA // Adjust for BCD count

JNC LOOP1

INR D

LOOP1: DCR C

JNZ LOOP2

STA C051 // Store the Least Significant Byte

MOV A,D

STA C052 // Store the Most Significant Byte

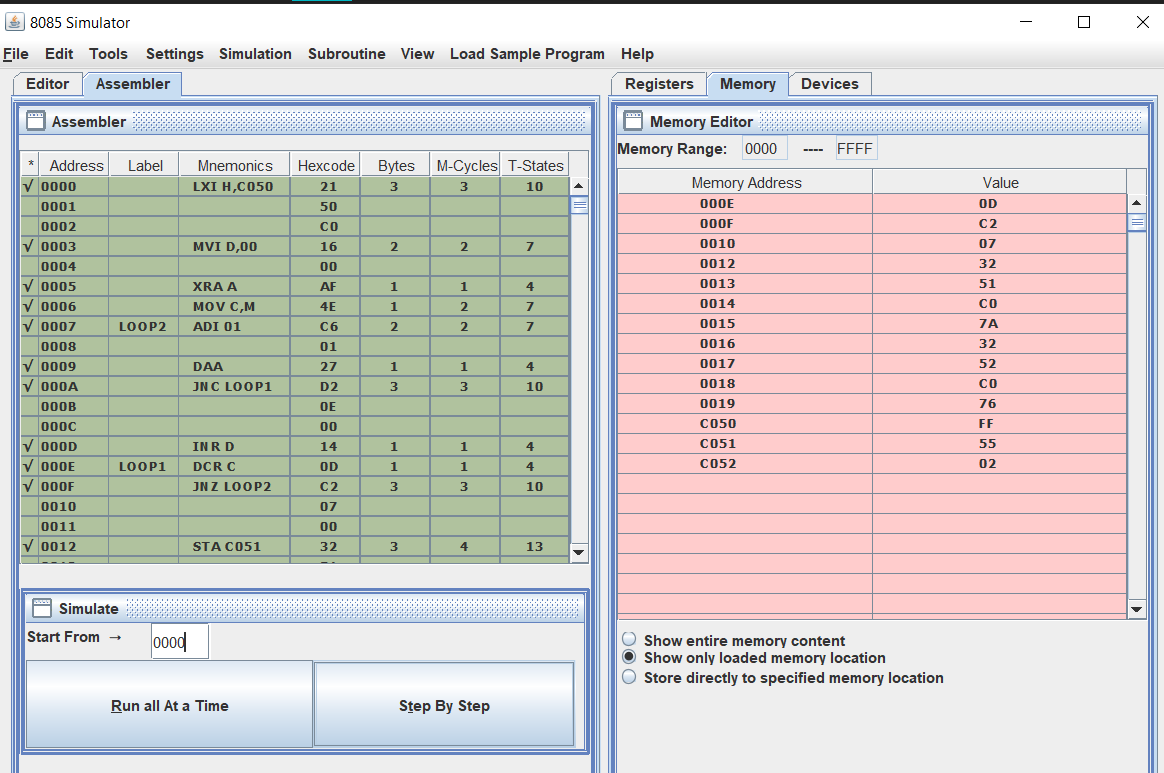
HLT

// HEX TO BCD

# ORG C050H

# DB FF

**Output:**



1. **Write a 8085 program to convert a BCD number into hexadecimal number.**

***Algo:***

1)Initialize memory pointer to 4150H location.

2) Get the Most Significant Digit (MSD).

3)Multiply the MSD by ten using repeated addition

4) Add the Least Significant Digit (LSD) to the result obtained in previous step.

5) Store the HEX data in Memory.

***Program:***

LXI H,C050H

MOV A,M

ADD A

MOV B,A

ADD A

ADD A

ADD B

INX H

ADD M

INX H

MOV M,A

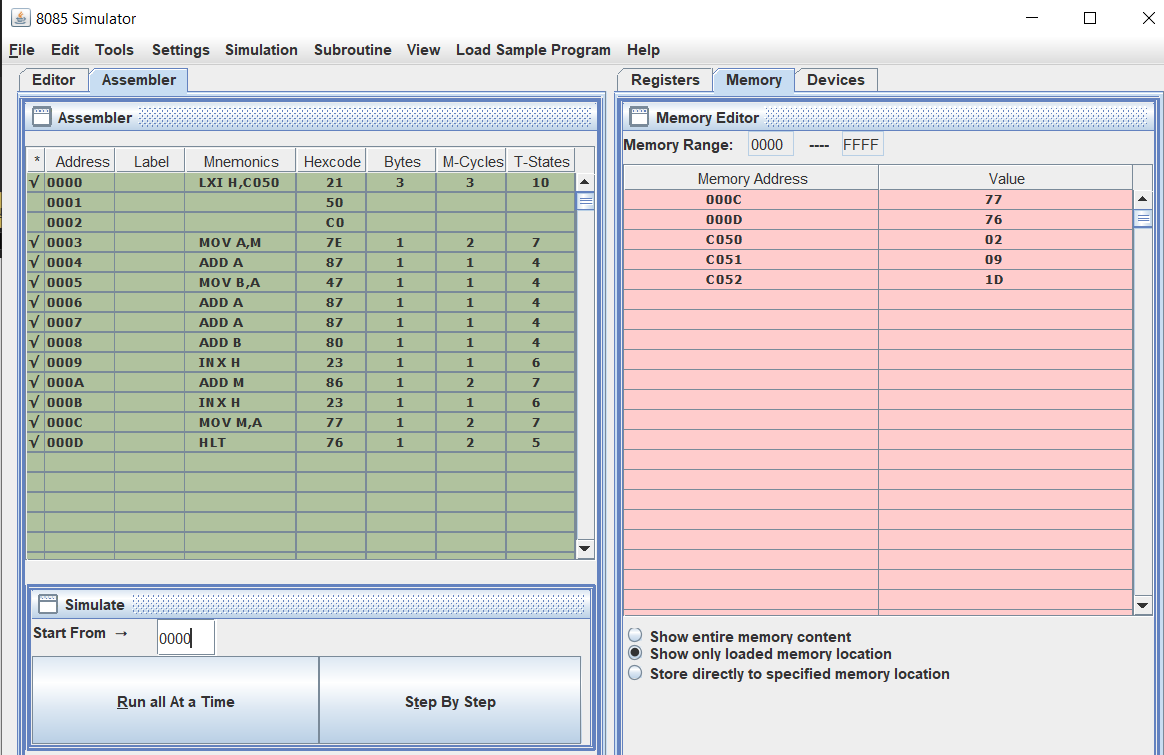
HLT

//BCD TO HEX

#ORG C050H

#DB 02H,09H

**Output:**

****