**Worksheet 3.3 or 10**

**Student Name:** Vivek Kumar **UID:** 21BCS8129

**Branch:** BE-CSE (LEET) **Section/Group:** ON20BCS-809/A

**Semester:** 4th Sem **Date of Performance:** 22/04/2022

**Subject Name:** MPI Lab **Subject Code:** 22E-20CSP-253

**1. Aim/Overview of the practical:**

1. Find the smaller number in an array.
2. Find the larger number in an array.

**2. Task to be done:**

Write an 8085 Microprocessor program to find the smaller & larger number in an array.

**3. Apparatus/Simulator used (For applied/experimental sciences/materials-based labs):**

1. 8085 Jubin simulator version 2 (Microprocessor Simulator)
2. Java (jdk/ jre1.8.0\_321)

**4. Algorithm/Flowchart (For programming-based labs):**

**Algorithm to Find the smaller out of two numbers:**

1. Load the Counter data to Memory from 1000 address using Immediate Instruction **LXI H, 1000**.
2. Move The data from Memory 1000 to Register ‘C’.
3. Increment the HL pair using **INX H** which will access the data from next address.
4. Move The data from Memory ‘M’ to Accumulator ‘A’.
5. Decrease the counter value by 1 which is stored in Register ‘C’ using **DCR C**.
6. Increment the HL pair using **INX H** which will access the data from next address.
7. Compare Data stored in Memory ‘M’ with Accumulator using **CMP M**.
8. Check if Carry flag generated using **JC** Instruction.
9. If carry flag generated and set to 1 jump to the label and decrease the counter value by 1 which is stored in Register ‘C’ using **DCR C**.
10. If carry flag not generated and set to 0. Move The data from Memory ‘M’ to Accumulator ‘A’ and then decrease the counter value by 1 which is stored in Register ‘C’ using **DCR C**.
11. Now check with the Zero flag generated or not using **JNZ** instruction.
12. If zero flag not generated and set to 0 jump to the loop and increment HL pair using **INX H** which will access the data from next address.
13. Store the Accumulator data to the To the Memory address 2000 using **STA 2000.**
14. End the execution using HLT.

**Algorithm to Find the larger out of two numbers:**

1. Load the Counter data to Memory from 1000 address using Immediate Instruction **LXI H, 1000**.
2. Move The data from Memory 1000 to Register ‘C’.
3. Increment the HL pair using **INX H** which will access the data from next address.
4. Move The data from Memory ‘M’ to Accumulator ‘A’.
5. Decrease the counter value by 1 which is stored in Register ‘C’ using **DCR C**.
6. Increment the HL pair using **INX H** which will access the data from next address.
7. Compare Data stored in Memory ‘M’ with Accumulator using **CMP M**.
8. Check if Carry flag not generated using **JNC** Instruction.
9. If carry flag no generated and set to 0 jump to the label and decrease the counter value by 1 which is stored in Register ‘C’ using **DCR C**.
10. If carry flag generated and set to 1. Move The data from Memory ‘M’ to Accumulator ‘A’ and then decrease the counter value by 1 which is stored in Register ‘C’ using **DCR C**.
11. Now check with the Zero flag generated or not using **JNZ** instruction.
12. If zero flag not generated and set to 0 jump to the loop and increment HL pair using **INX H** which will access the data from next address.
13. Store the Accumulator data to the To the Memory address 2000 using **STA 2000.**
14. End the execution using HLT.

**5. Description/ Code:**

**Program to Find the smaller number in an array:**

# ORG 0900H

LXI H,1000

MOV C,M

INX H

MOV A,M

DCR C

**LOOP**: INX H

CMP M

JC **LABEL**

MOV A,M

**LABEL**: DCR C

JNZ **LOOP**

STA 2000

HLT

# ORG 1000

# DB 06H,09H,F6H,09H,D5H,02H,10H

**Program to Find the larger number in an array:**

# ORG 0900H

LXI H,1000

MOV C,M

INX H

MOV A,M

DCR C

**LOOP**: INX H

CMP M

JNC **LABEL**

MOV A,M

**LABEL**: DCR C

JNZ **LOOP**

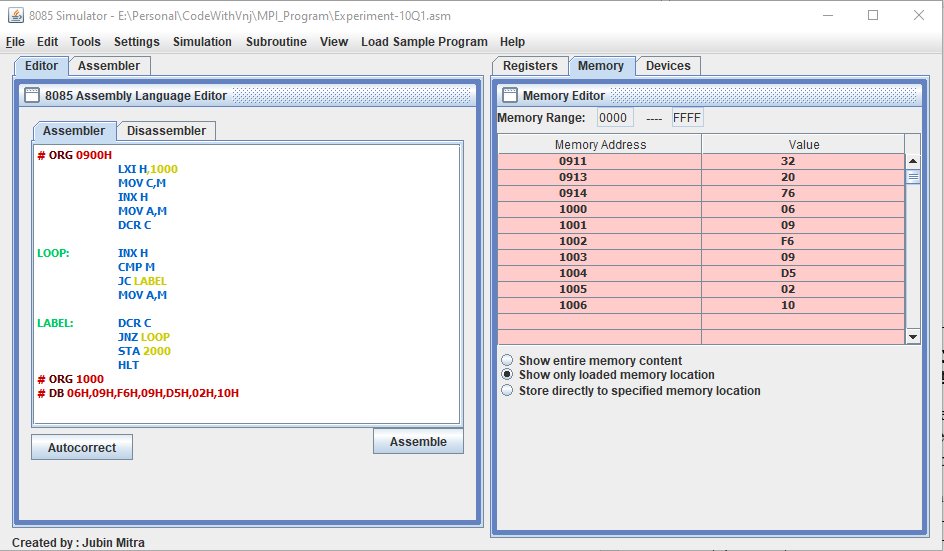
STA 2000

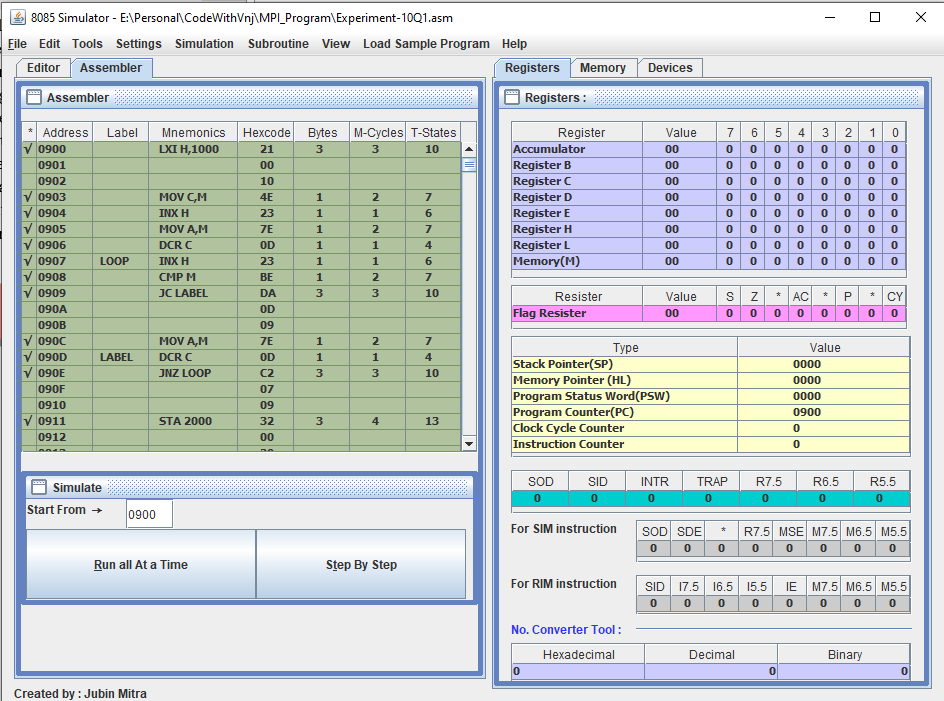
HLT

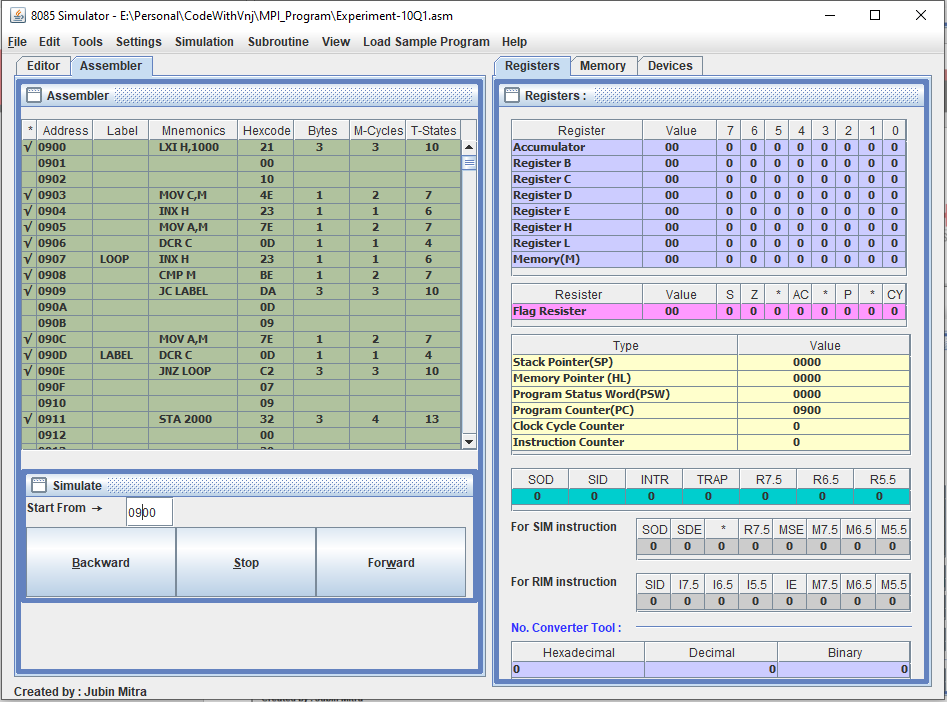
# ORG 1000

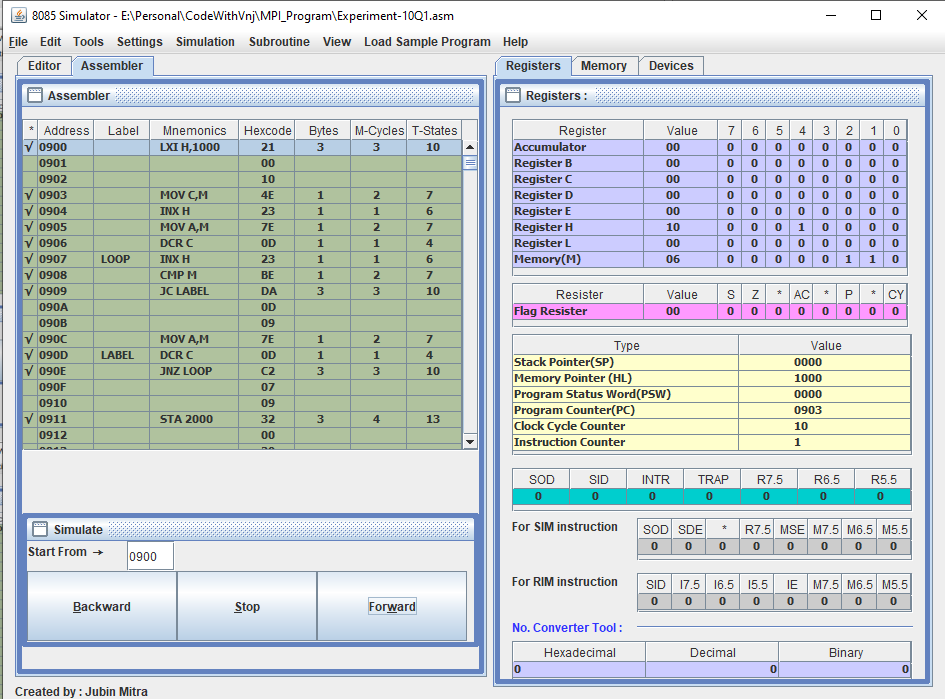
# DB 09H,01H,F6H,D0H,2DH,02H,F0H,FDH,A2H,F3H

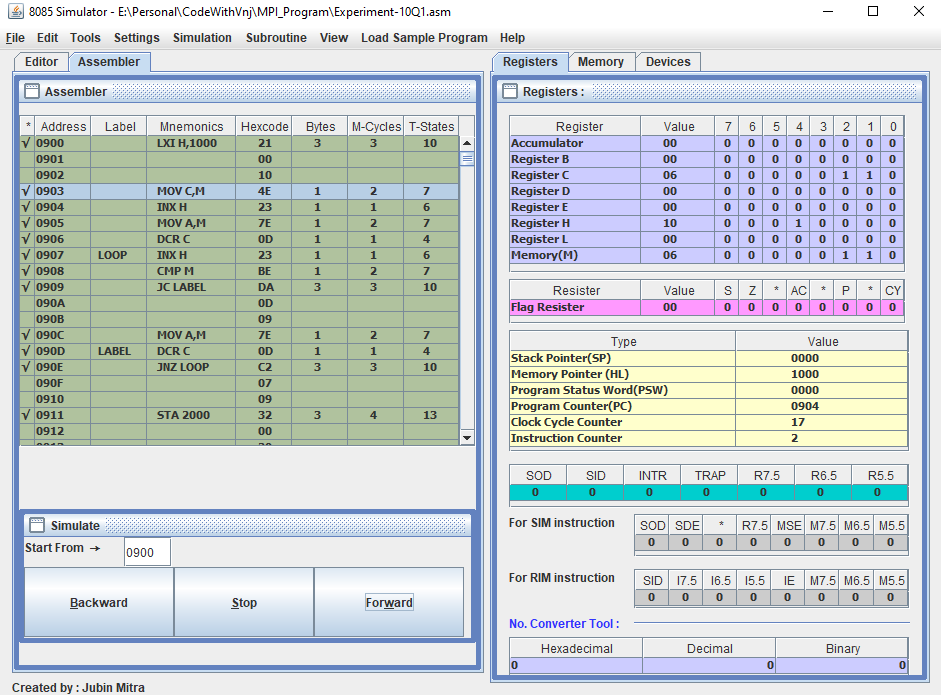
**6. Result/Output/Writing Summary:**

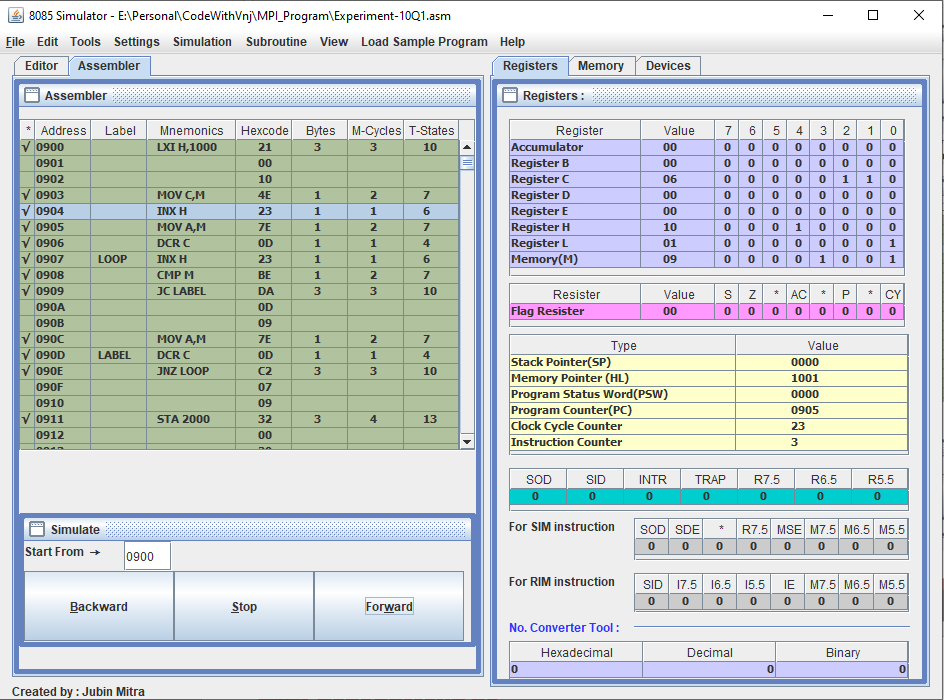
**Output to Find the smaller number in an array:**

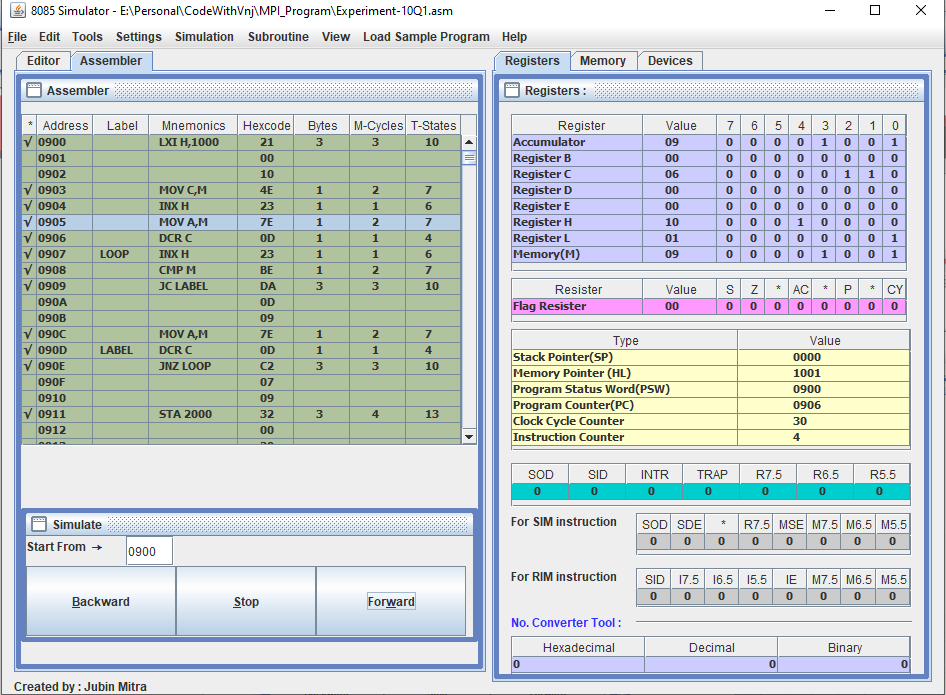


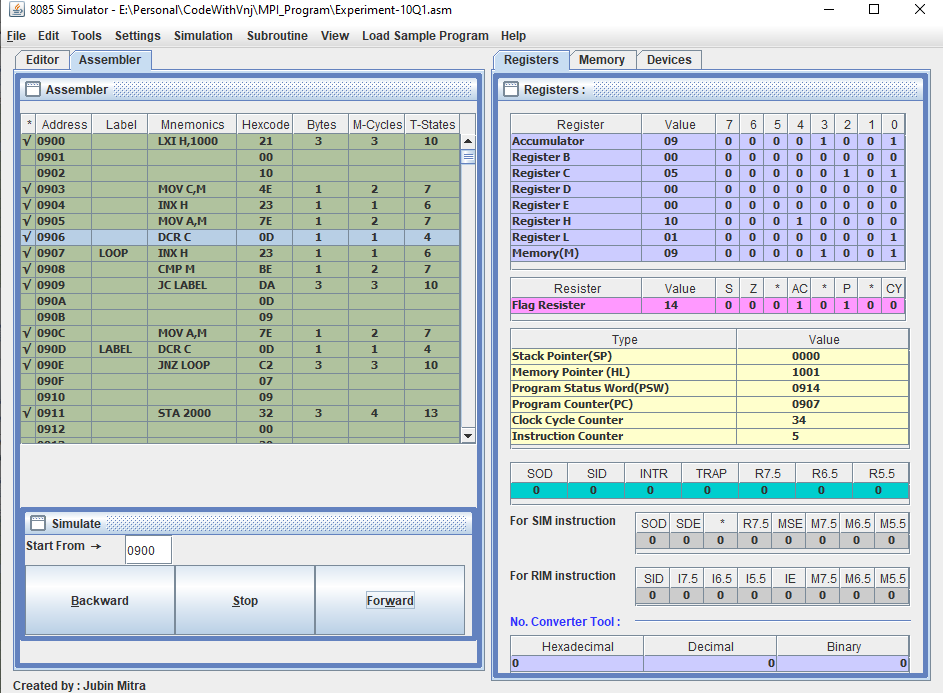


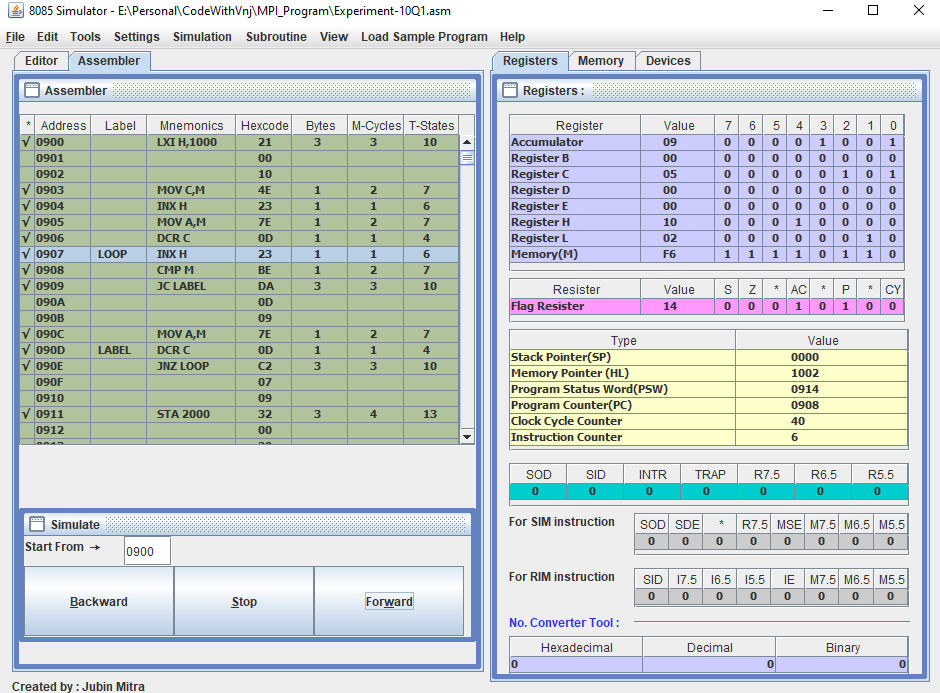


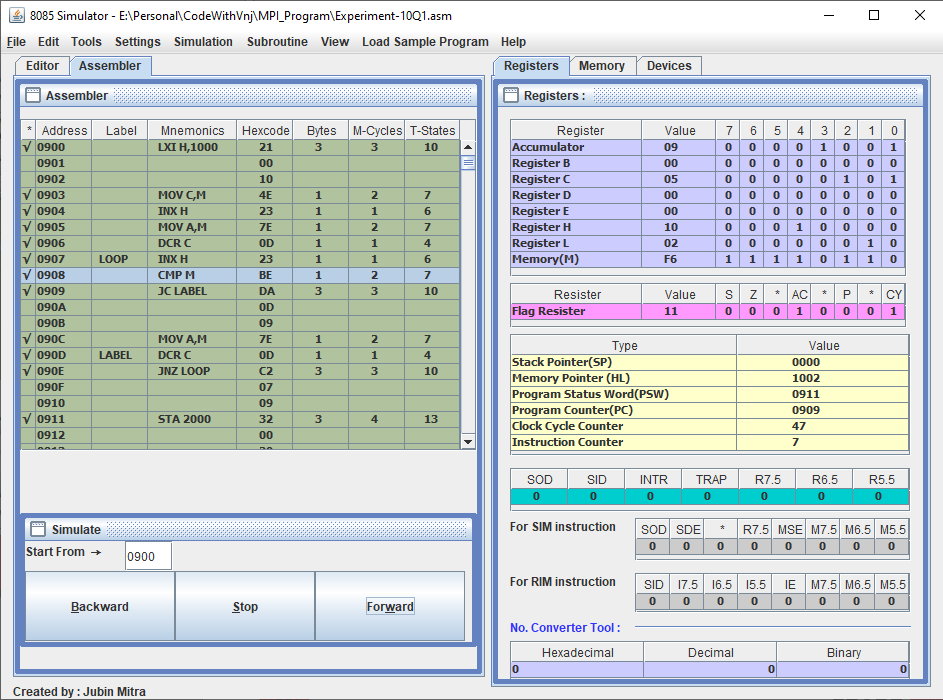


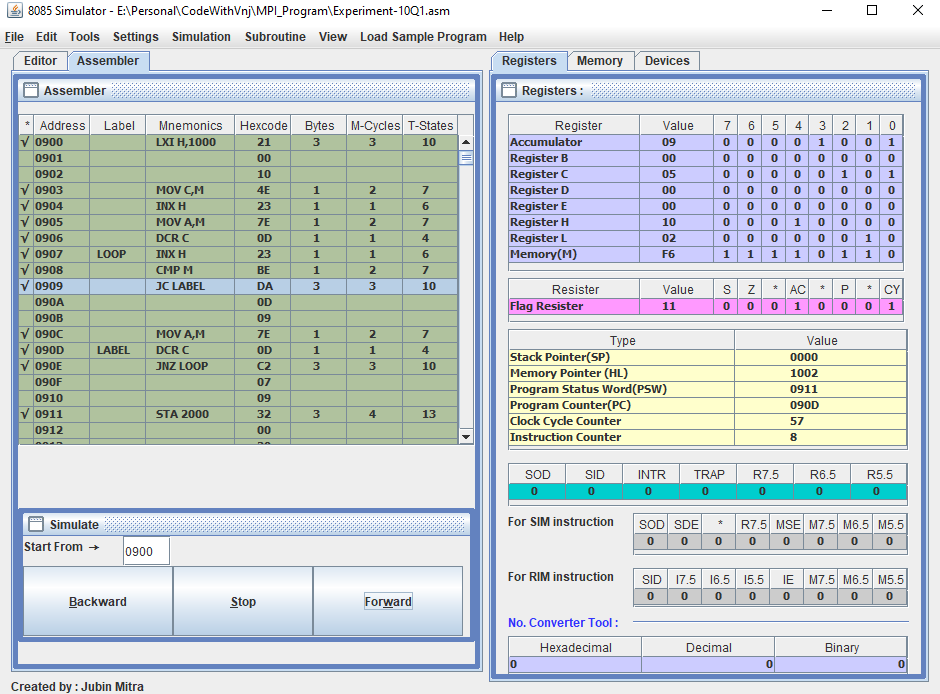


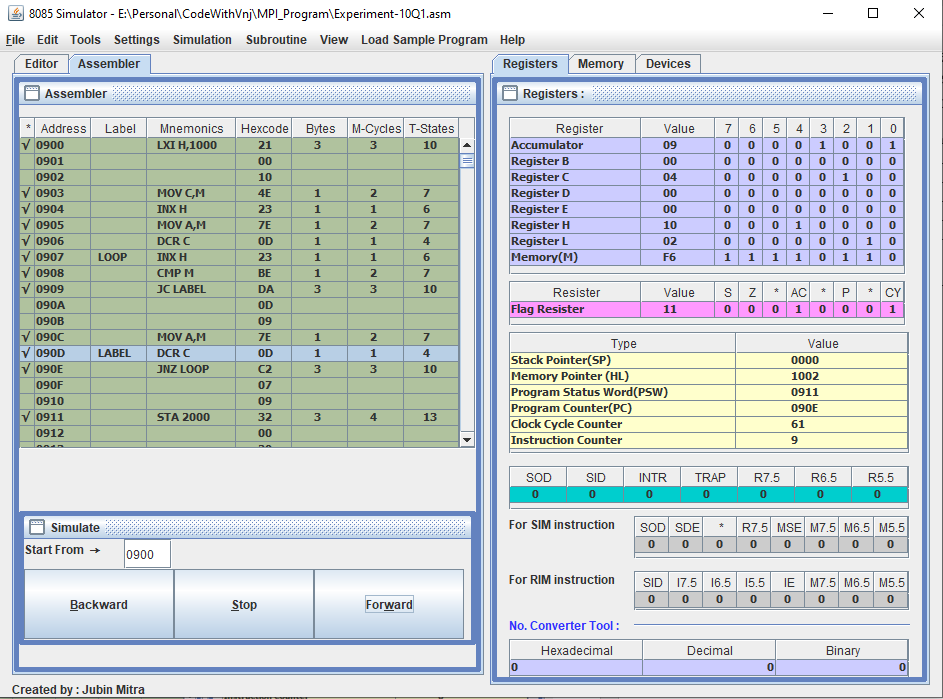


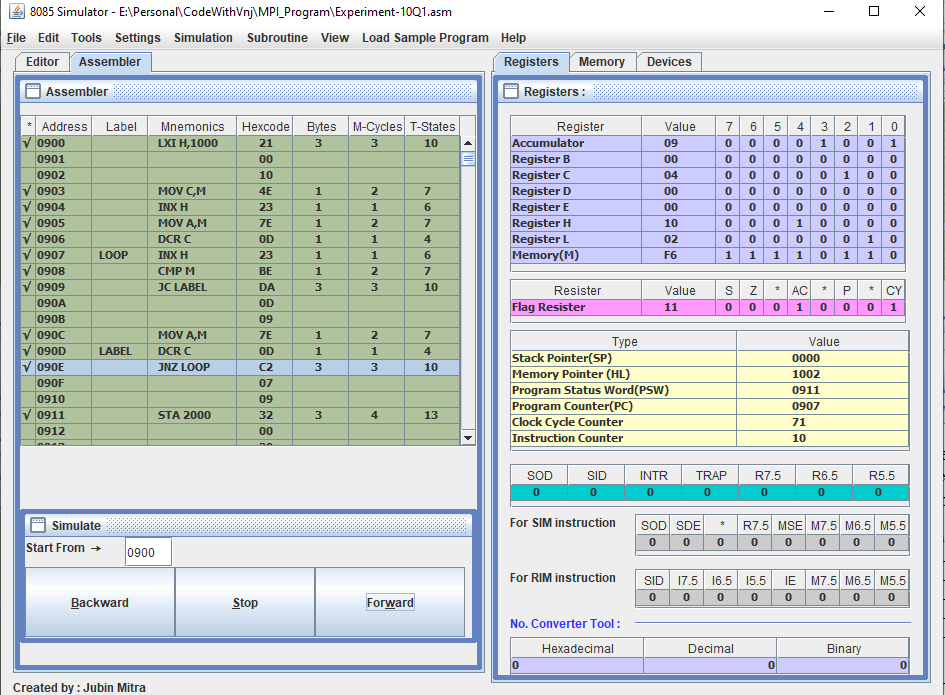




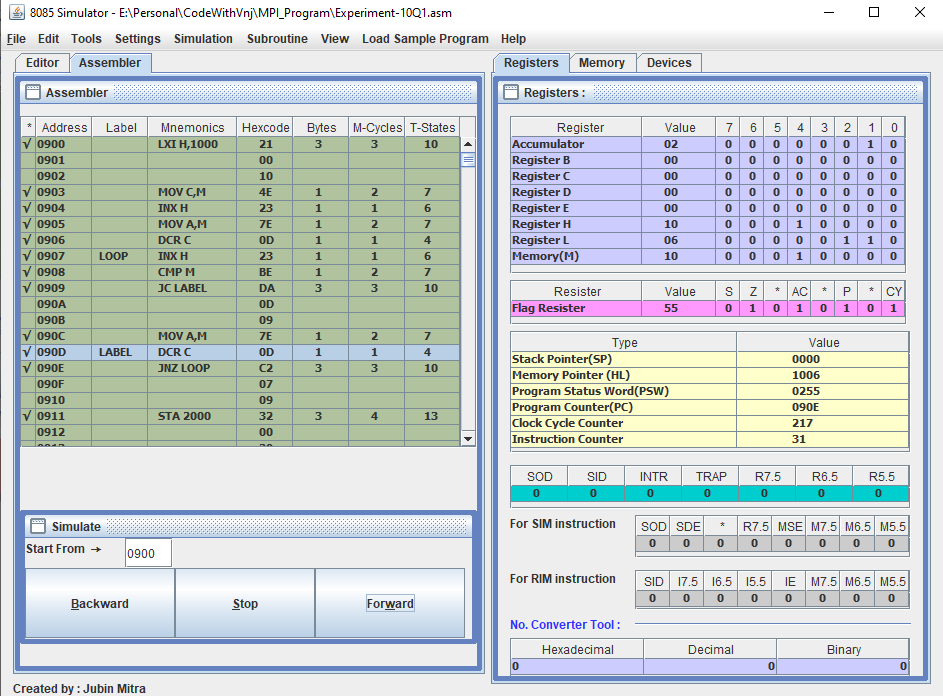


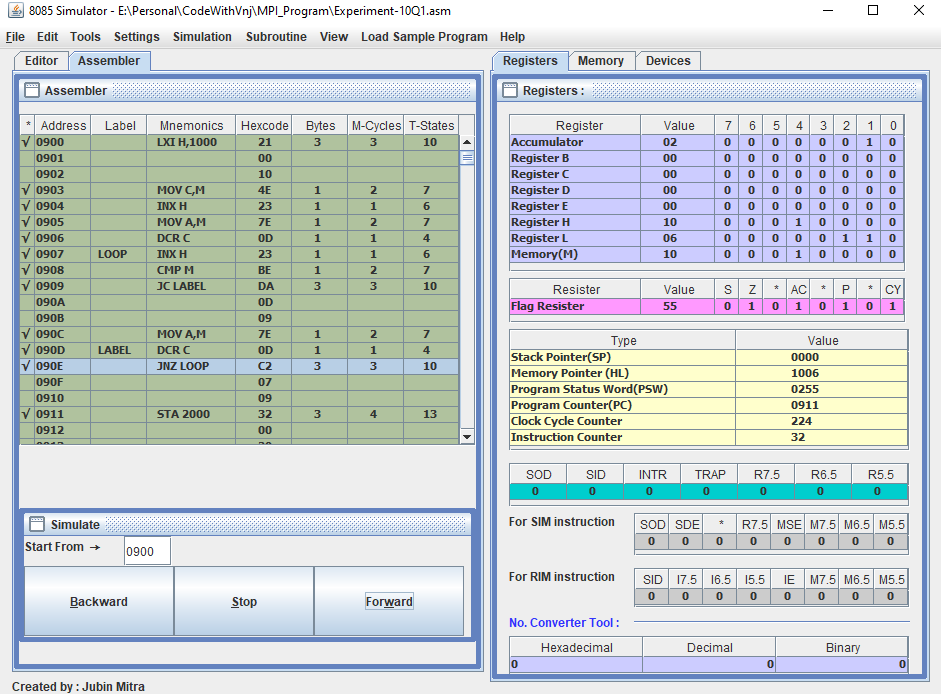


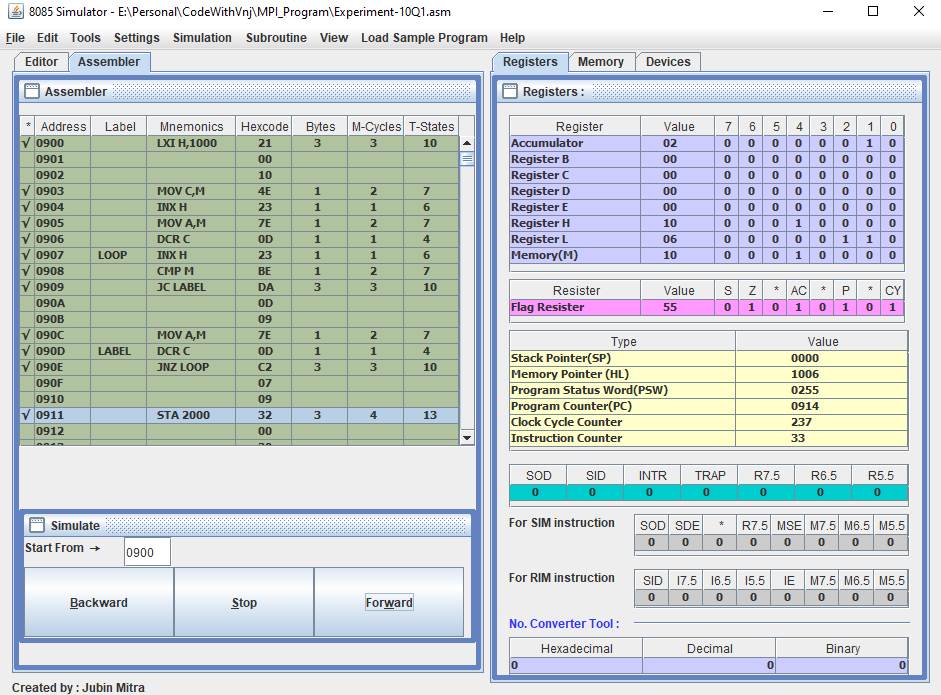


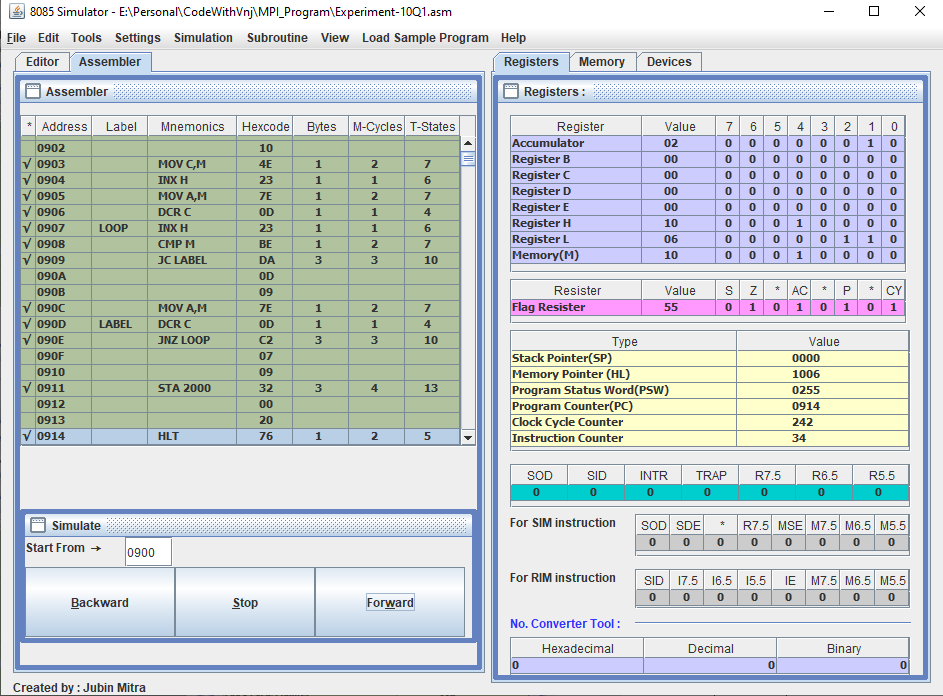


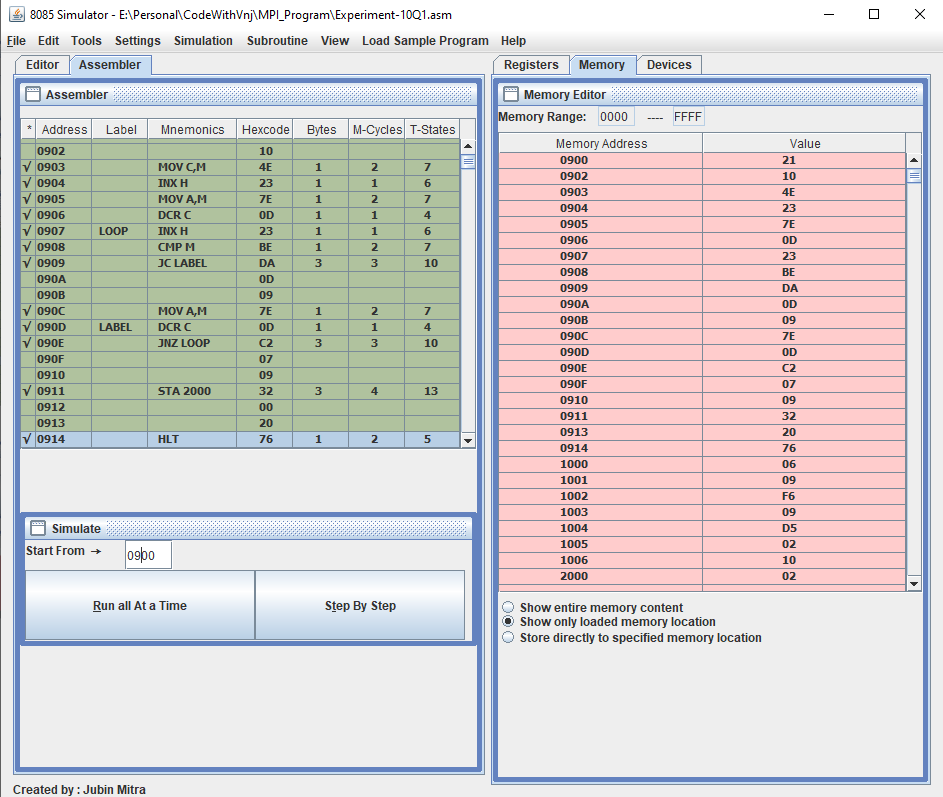
Now the loop will run Until our Zero flag set to 1. And then it will finally store the Accumulator value to 2000.



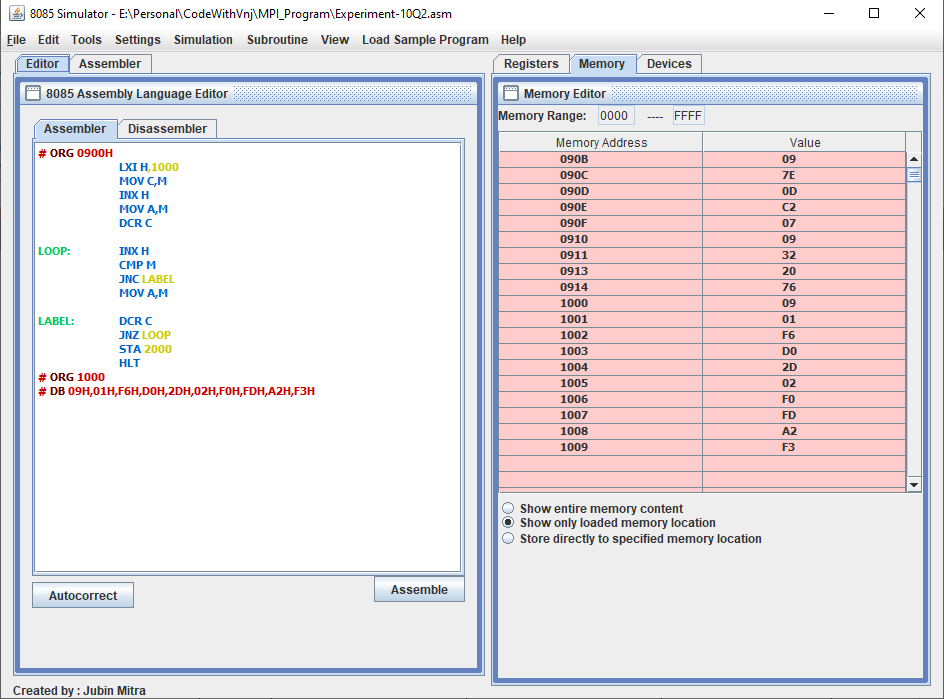


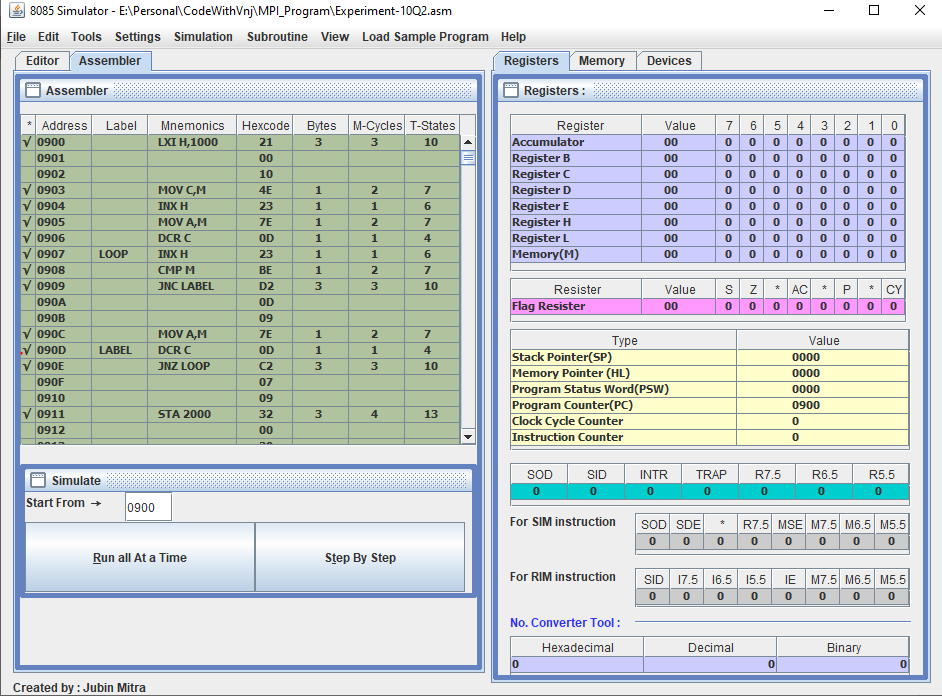


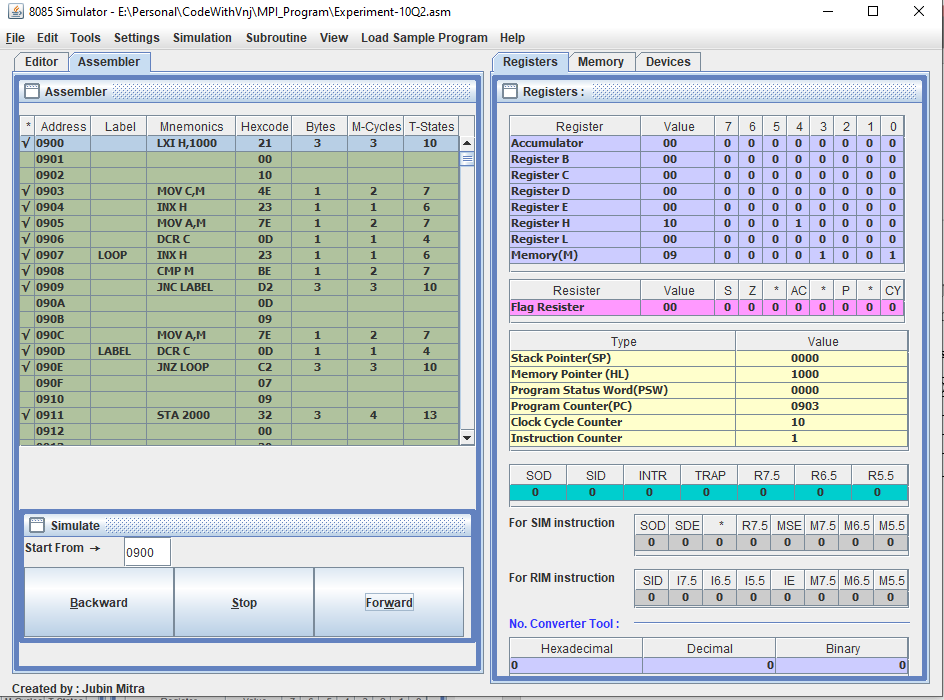


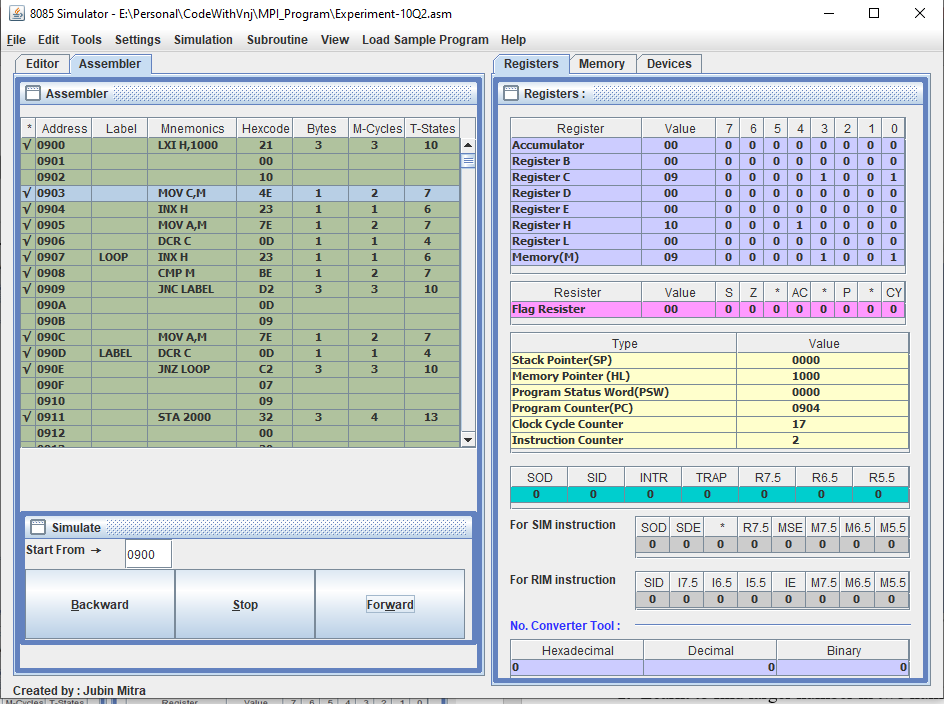


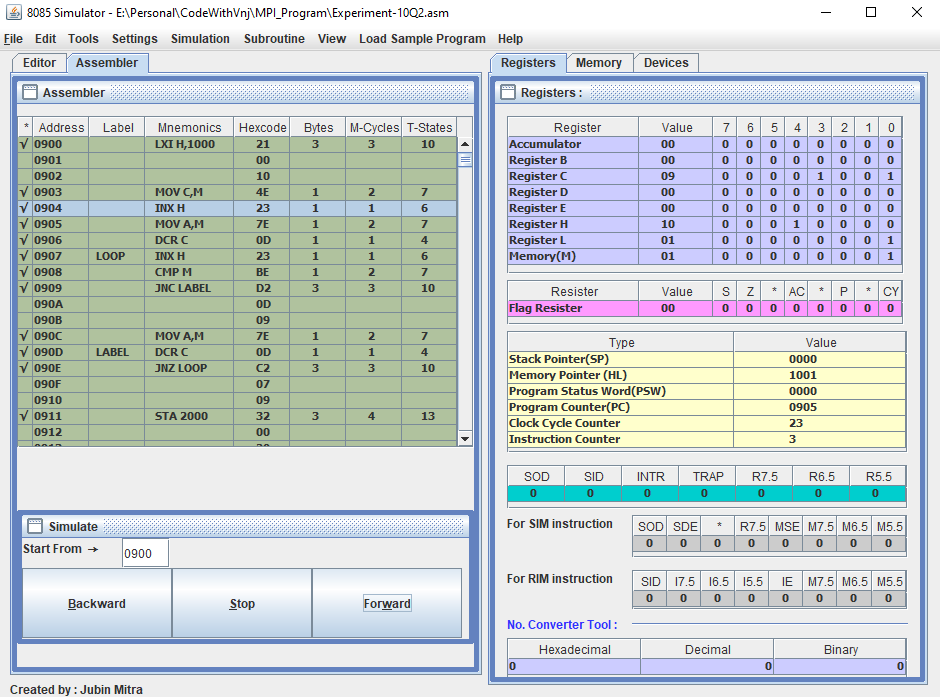
**Output to Find the larger number in an array:**

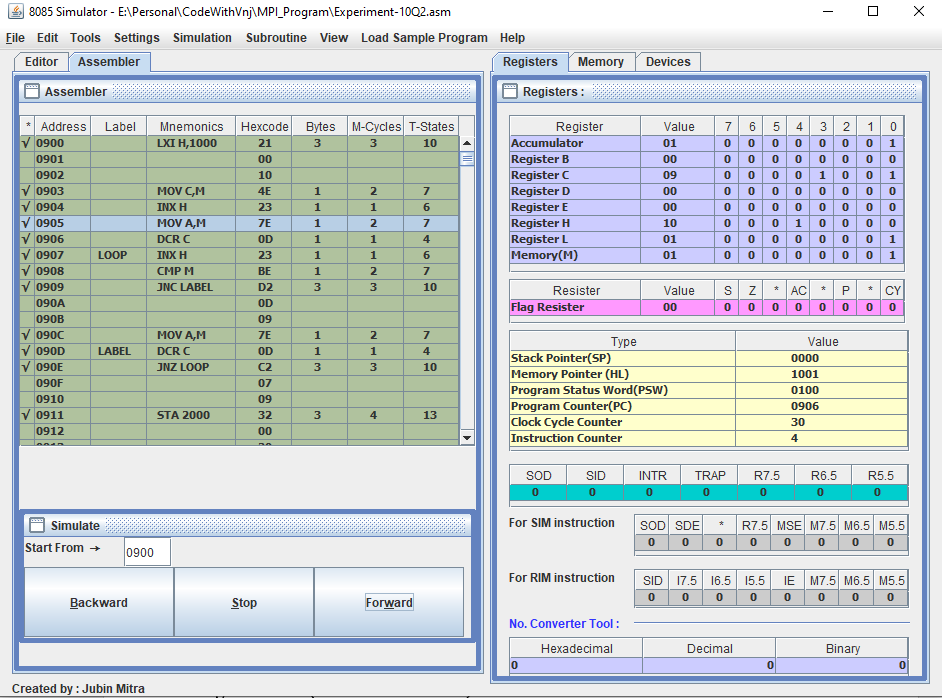


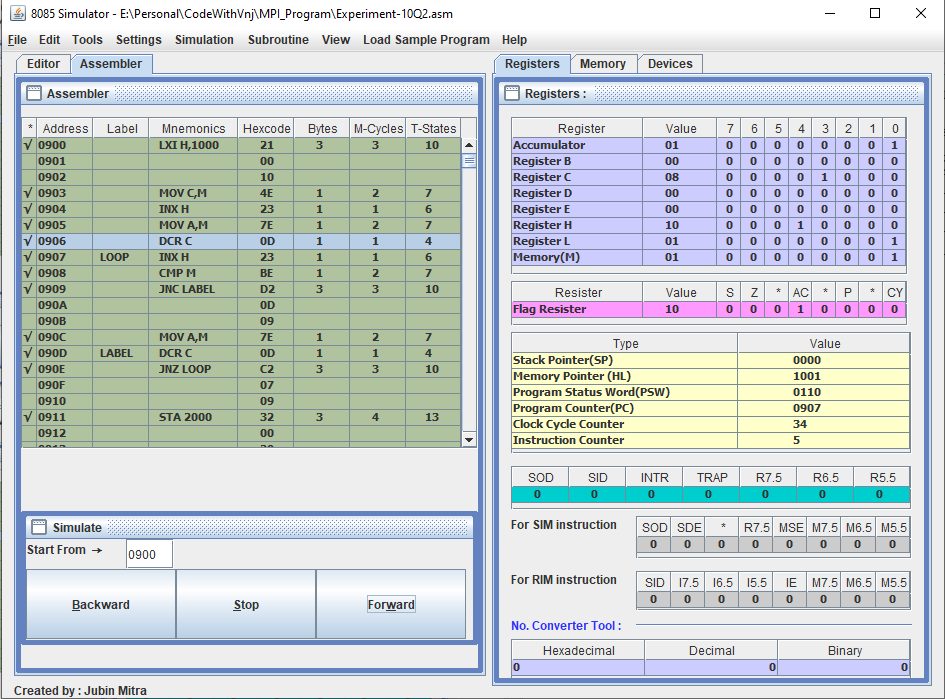


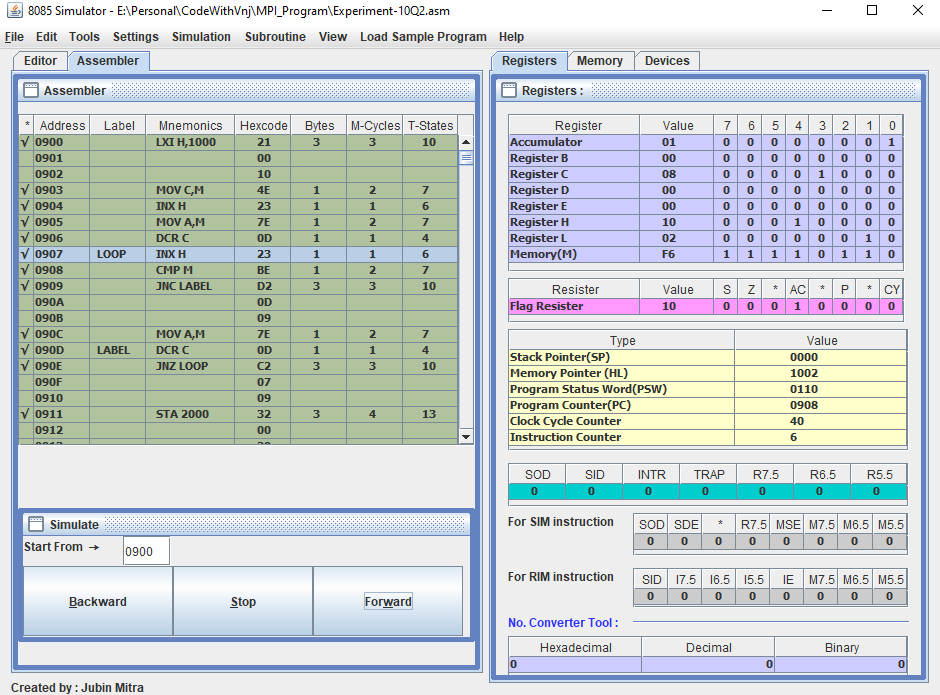


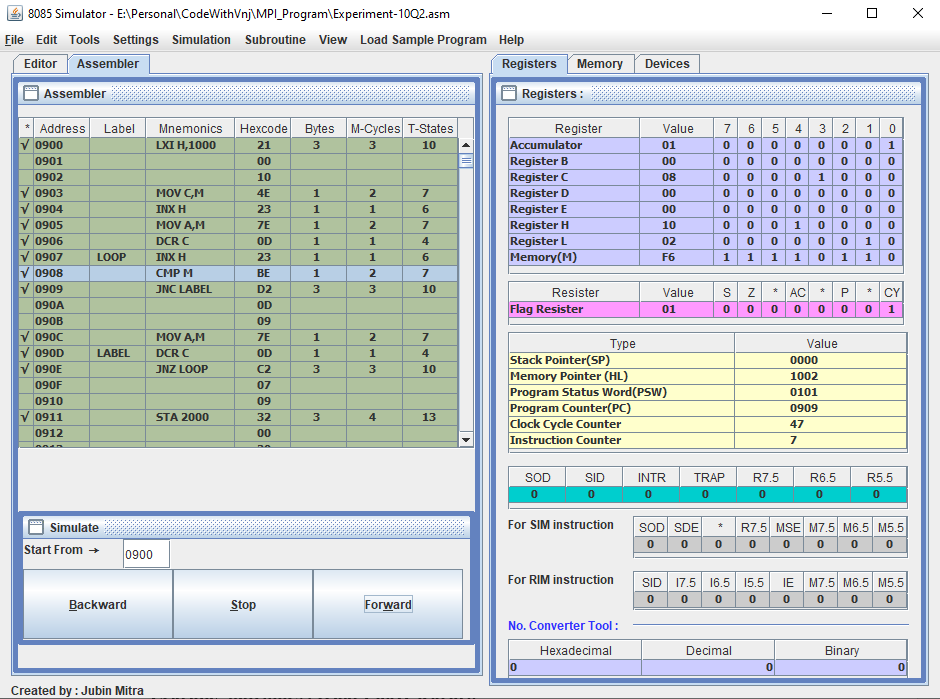


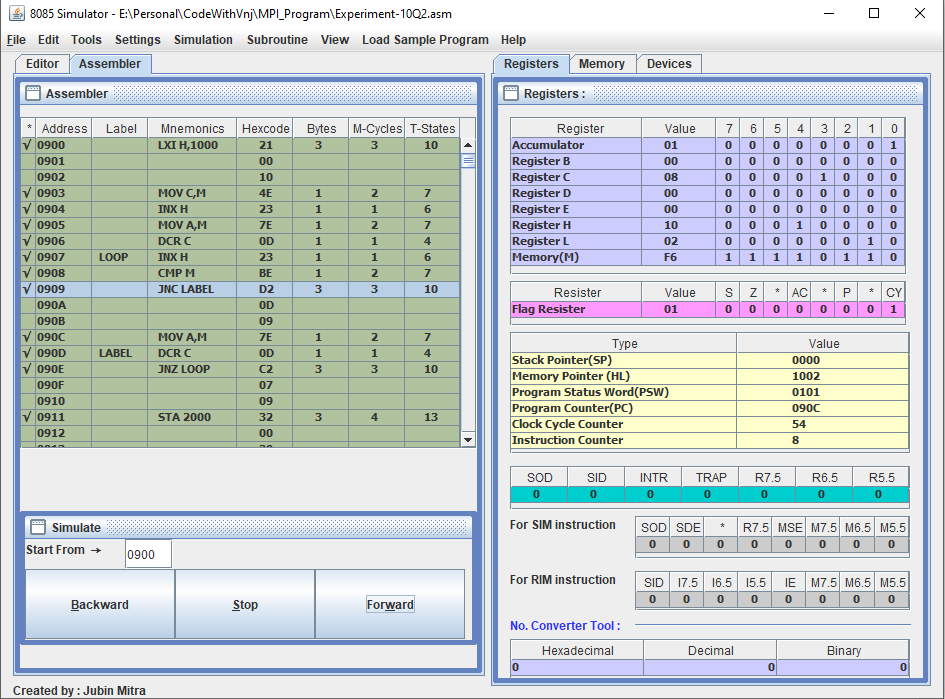


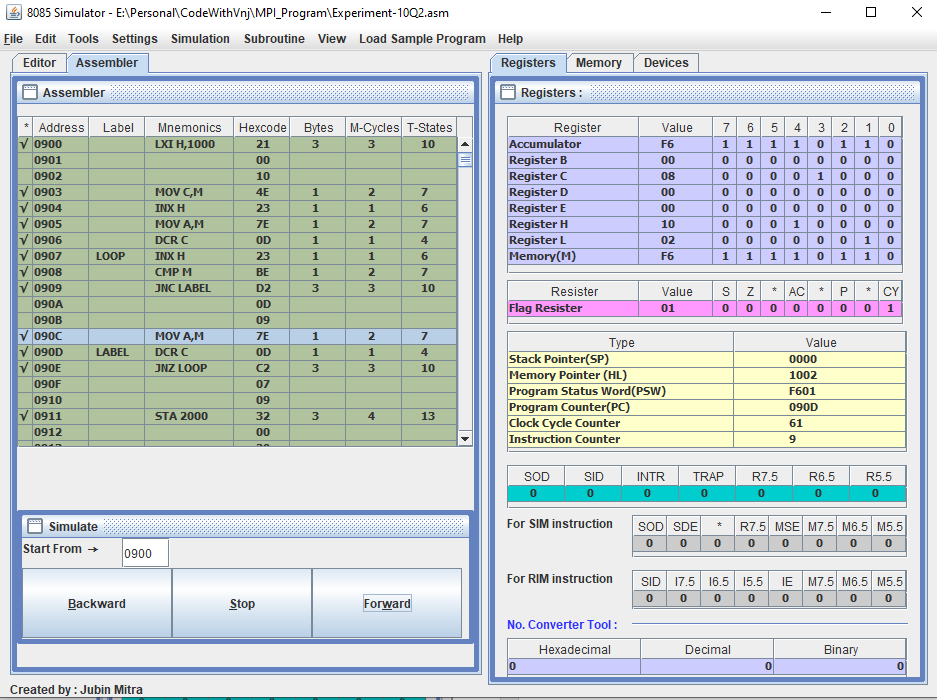


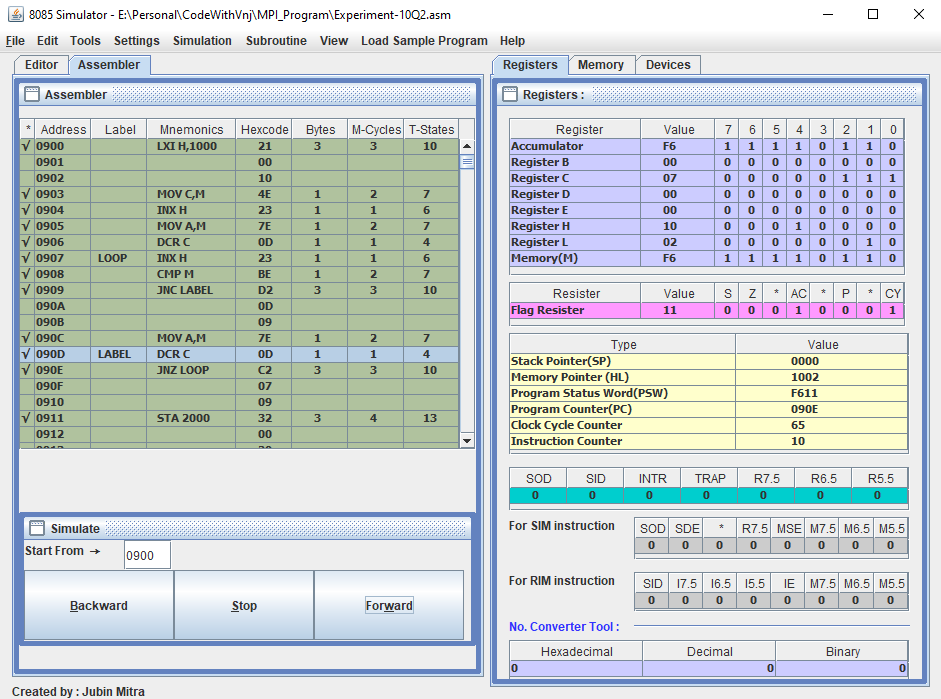


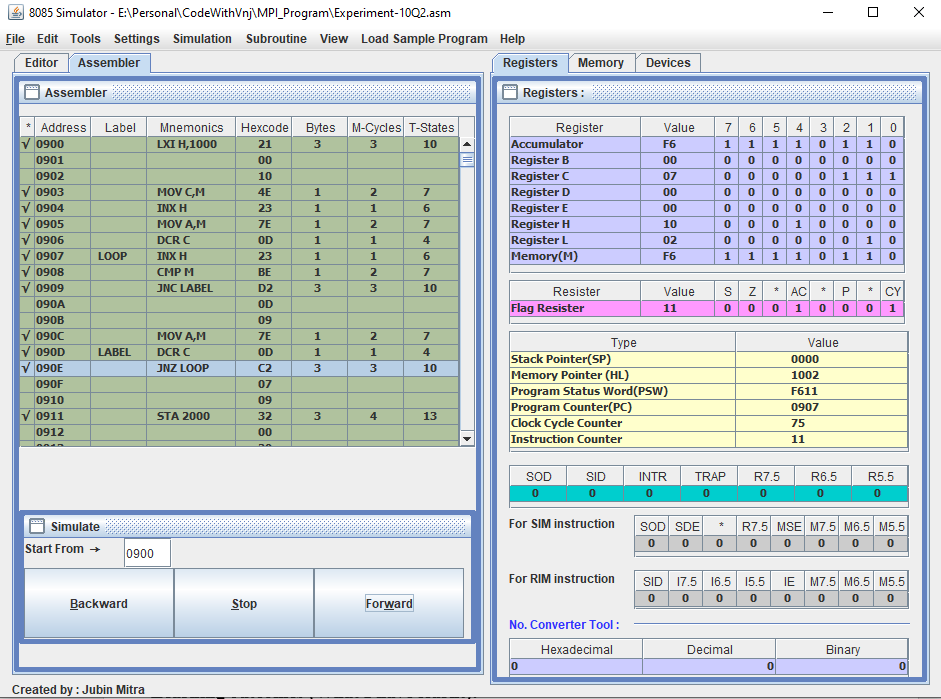




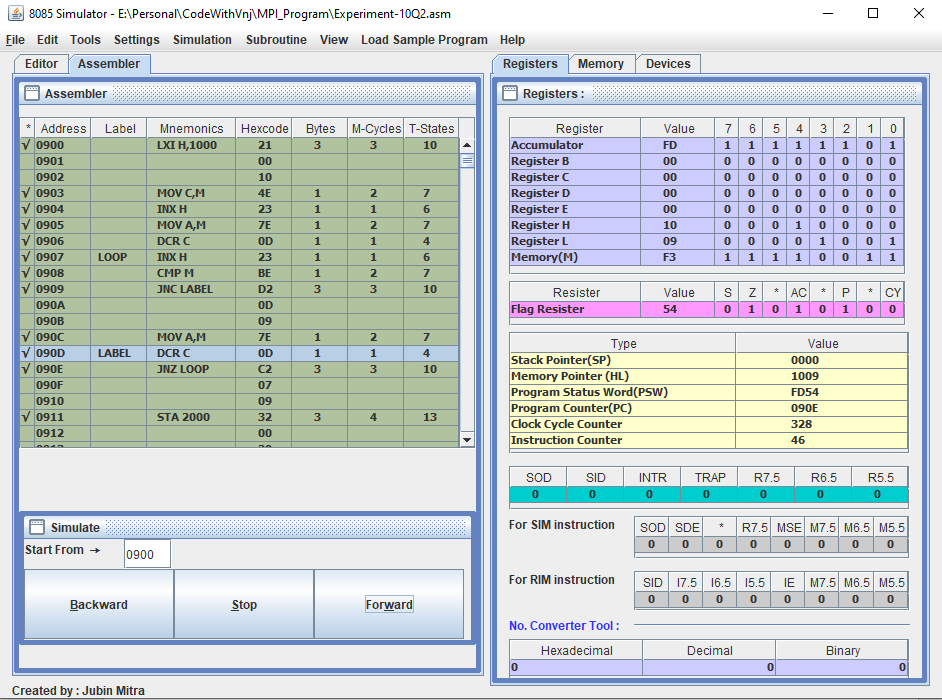


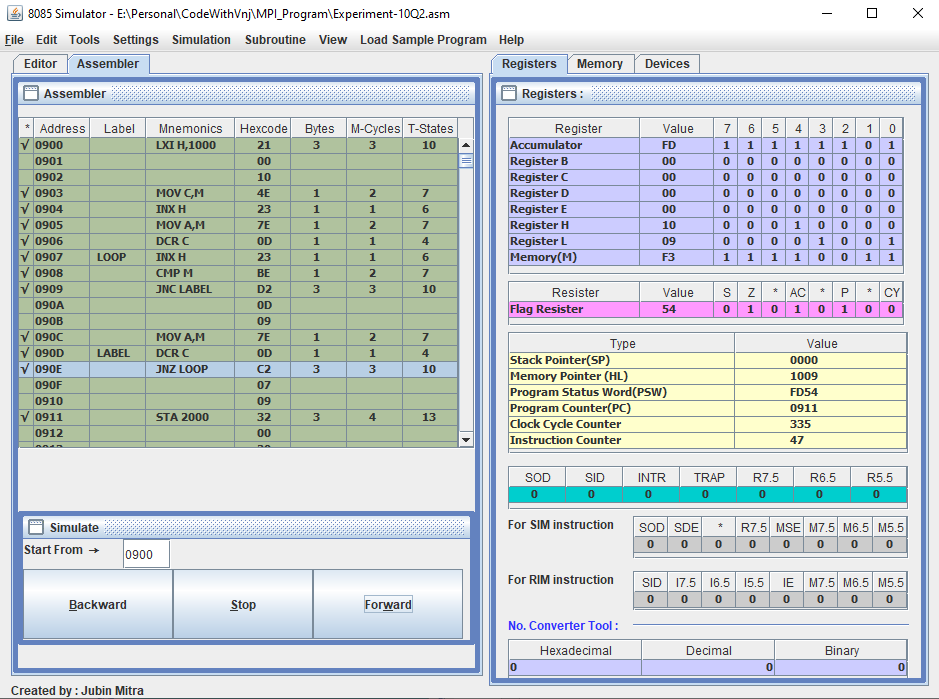
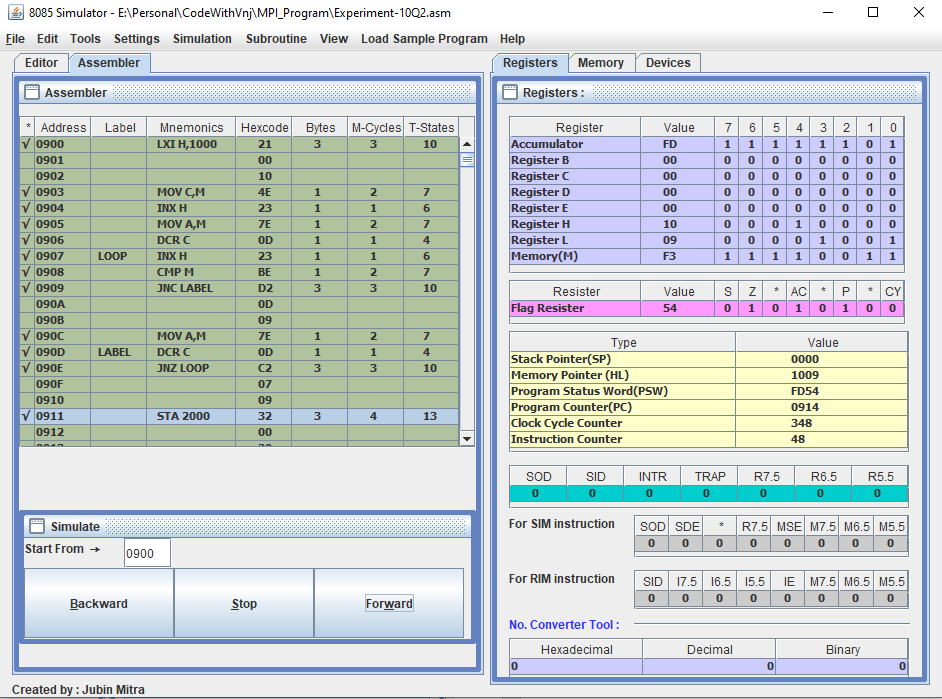


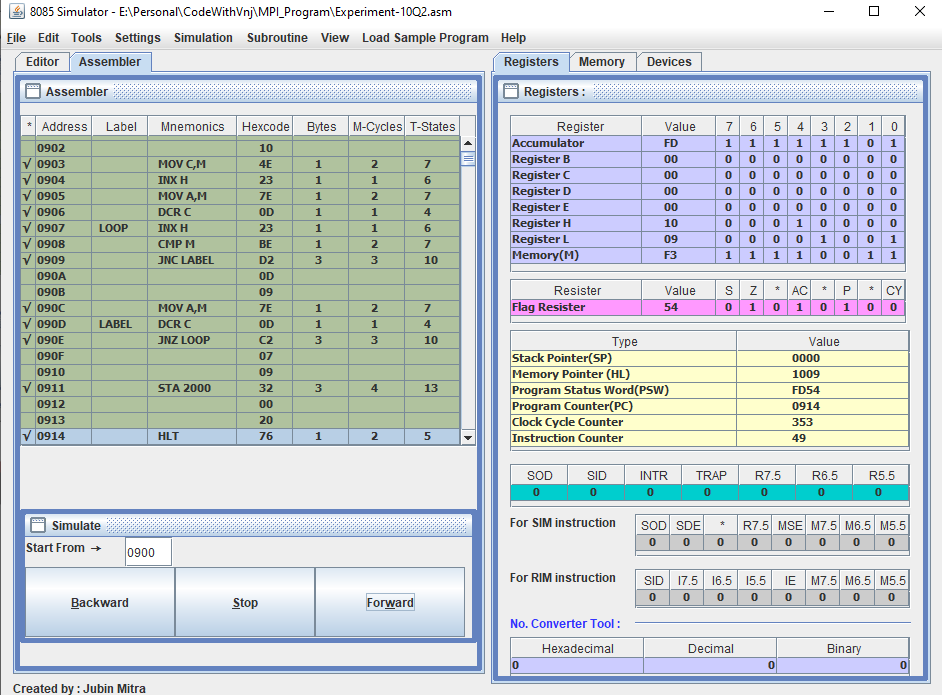


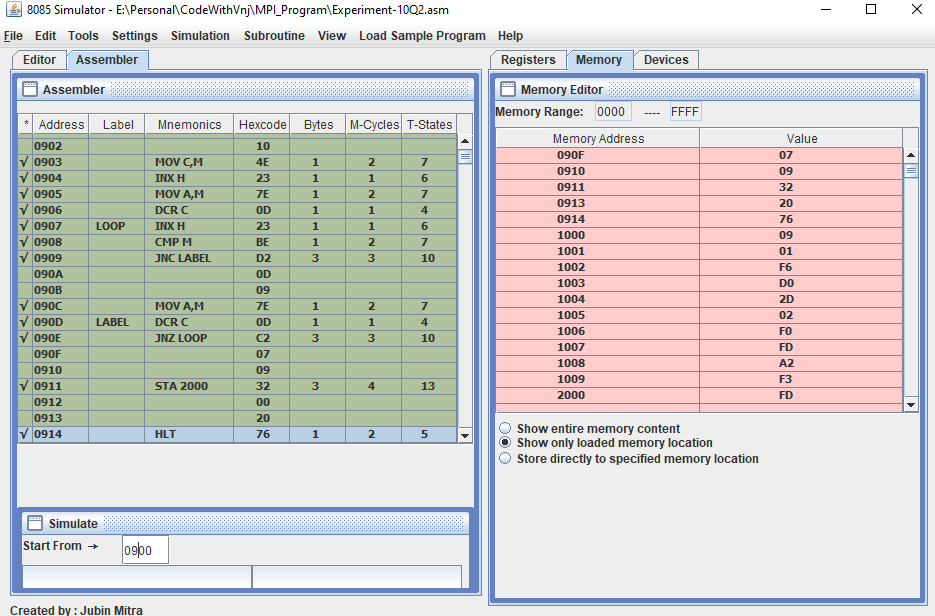


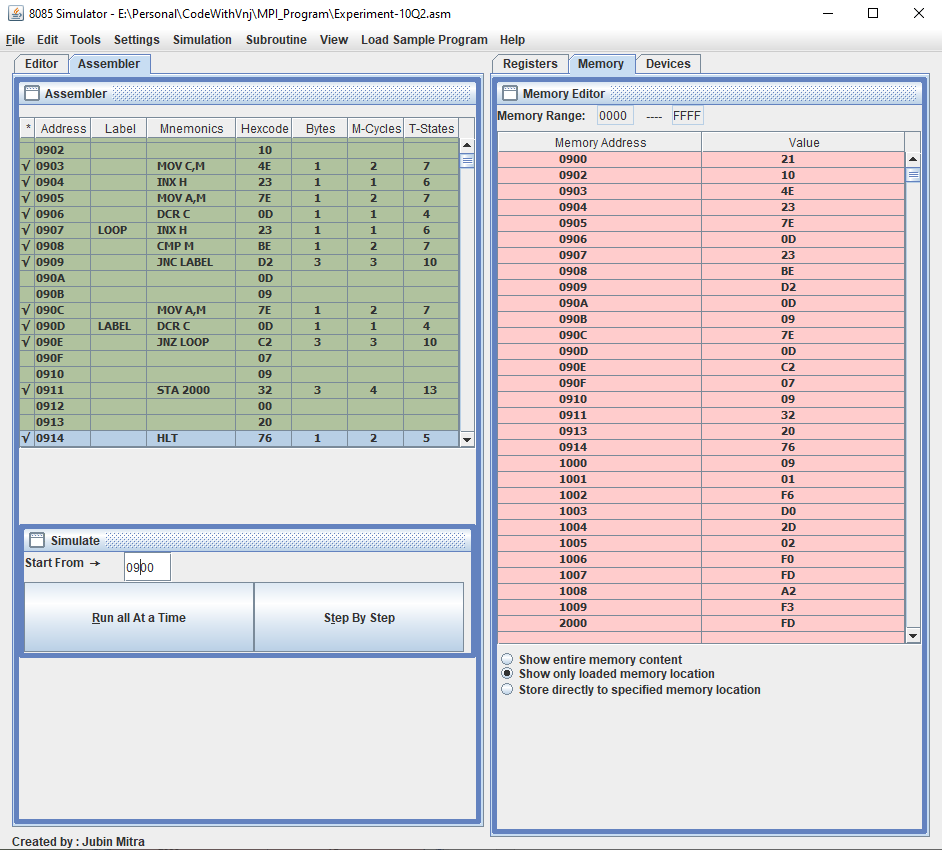
Now the loop will run Until our Zero flag set to 1. And then it will finally store the Accumulator value to 2000.











**Learning outcomes (What I have learnt):**

1. Learnt to find larger number in a given Array.
2. Learnt to find smaller number in a given Array.
3. Learnt to use the Nested Looping statement in the single program.
4. Learnt to use the JC, JNC & JNZ instruction.

**Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. No. | Parameters | Marks Obtained | Maximum Marks |
| 1. |  |  |  |
| 2. |  |  |  |
| 3. |  |  |  |
| 4. |  |  |  |