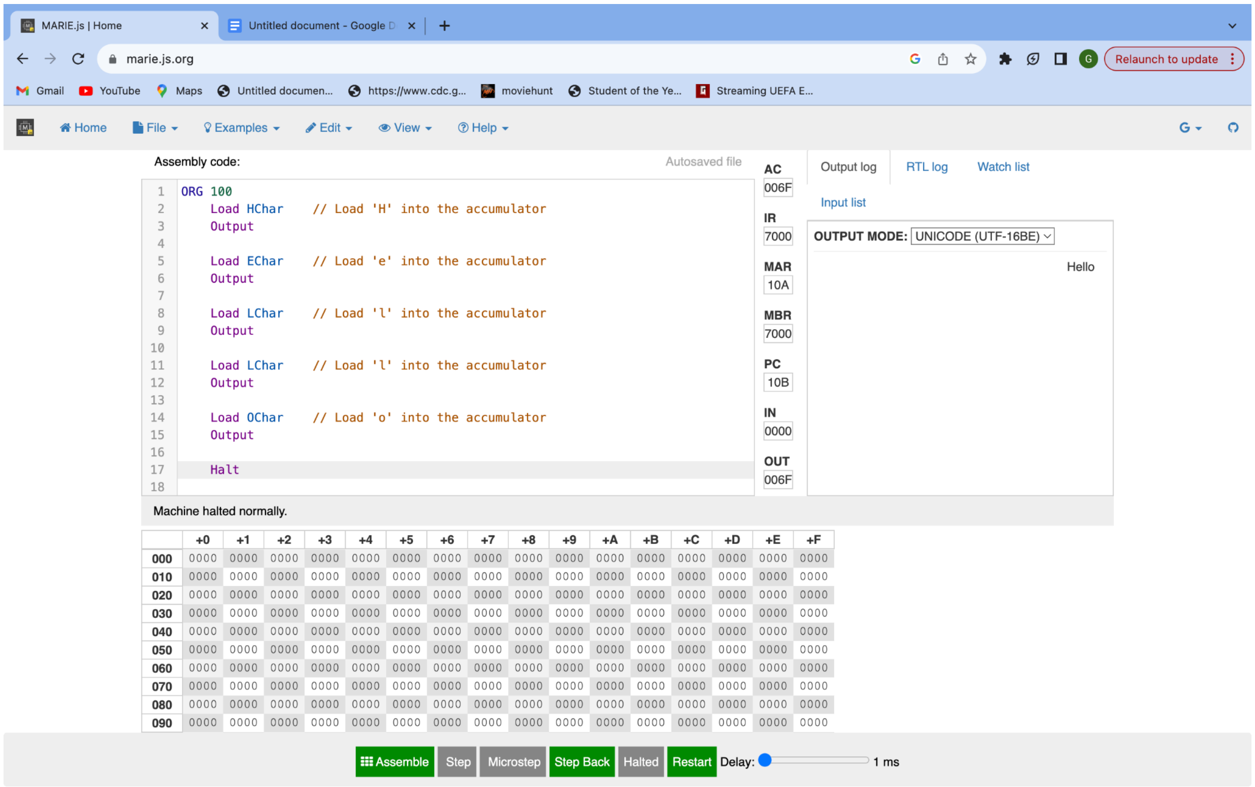
Qno.1)

Code:

```
ORG 100
   Load HChar    // Load 'H' into the accumulator
   Output

   Load EChar    // Load 'e' into the accumulator
   Output

   Load LChar    // Load 'l' into the accumulator
   Output

   Load LChar    // Load 'l' into the accumulator
   Output

   Load OChar    // Load 'o' into the accumulator
   Output

   Halt

HChar, HEX 48      // ASCII for 'H'
EChar, HEX 65      // ASCII for 'e'
LChar, HEX 6C      // ASCII for 'l'
OChar, HEX 6F      // ASCII for 'o'
```

Output:



Assembly code: Autosaved file

```
ORG 100
    Load HChar    // Load 'H' into the accumulator
    Output

    Load EChar    // Load 'e' into the accumulator
    Output

    Load LChar    // Load 'l' into the accumulator
    Output

    Load LChar    // Load 'l' into the accumulator
    Output

    Load OChar    // Load 'o' into the accumulator
    Output

    Halt

```

AC 006F
IR 7000
MAR 10A
MBR 7000
PC 10B
IN 0000
OUT 006F

OUTPUT MODE: UNICODE (UTF-16BE)

```
Hello
```

Machine halted normally.

	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
020	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
030	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
040	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
050	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
060	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
070	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
080	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
090	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

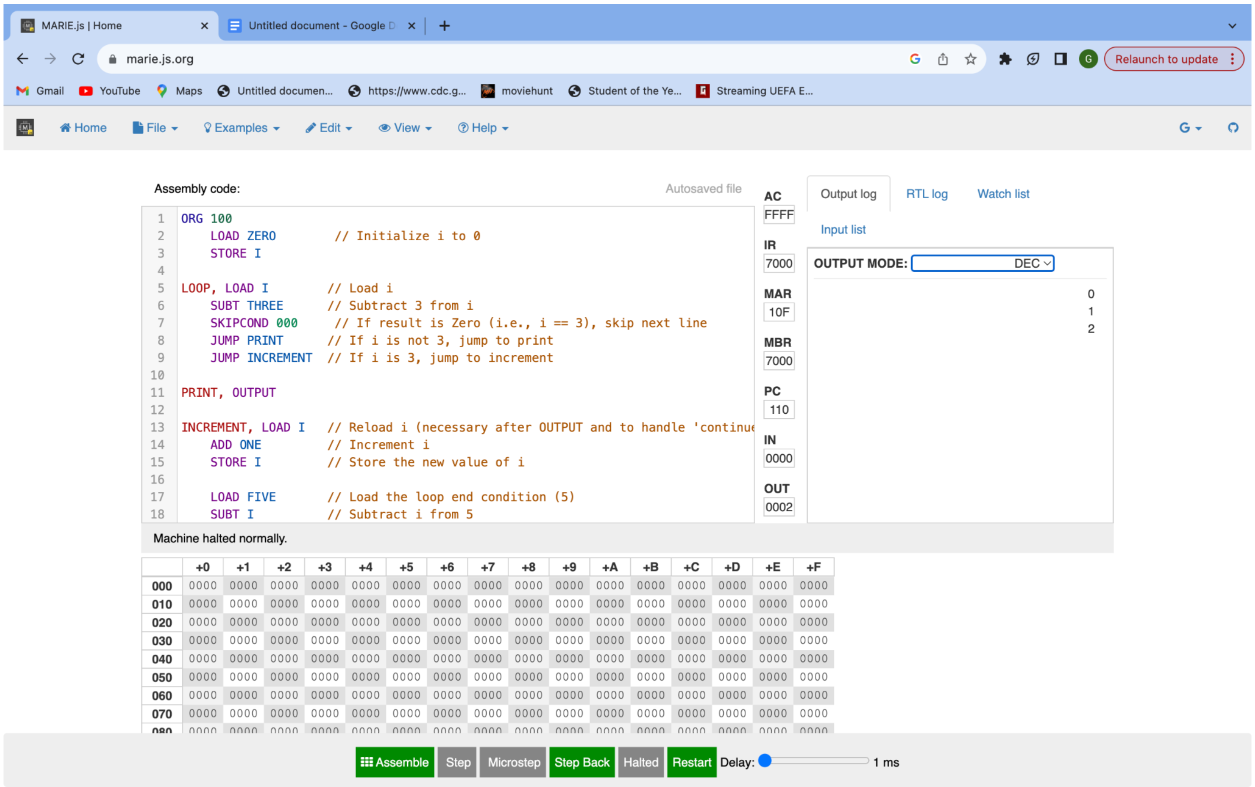
Delay: 1 ms

qno.2)

```
ORG 100
   LOAD ZERO        // Initialize i to 0
   STORE I

LOOP, LOAD I        // Load i
   SUBT THREE      // Subtract 3 from i
   SKIPCOND 000      // If result is Zero (i.e., i == 3), skip next line
   JUMP PRINT      // If i is not 3, jump to print
   JUMP INCREMENT  // If i is 3, jump to increment

PRINT, OUTPUT

INCREMENT, LOAD I   // Reload i (necessary after OUTPUT and to handle 'continue')
   ADD ONE         // Increment i
   STORE I         // Store the new value of i

   LOAD FIVE       // Load the loop end condition (5)
   SUBT I          // Subtract i from 5
```

```
    SKIPCOND 000      // If result is Negative (i.e., i >= 5), skip next line
    JUMP LOOP        // If i is less than 5, jump back to the start of the loop

ENDLOOP, HALT

I, DEC 0            // Memory location for loop variable i
ONE, DEC 1          // Constant 1
THREE, DEC 3         // Constant 3 (condition for continue)
FIVE, DEC 5         // Constant 5 (loop end condition)
ZERO, DEC 0         // Constant 0
```

Output:



```
ORG 100
    LOAD ZERO        // Initialize i to 0
    STORE I

LOOP, LOAD I        // Load i
    SUBT THREE      // Subtract 3 from i
    SKIPCOND 000     // If result is Zero (i.e., i == 3), skip next line
    JUMP PRINT      // If i is not 3, jump to print
    JUMP INCREMENT  // If i is 3, jump to increment

PRINT, OUTPUT

INCREMENT, LOAD I   // Reload i (necessary after OUTPUT and to handle 'continue

    ADD ONE         // Increment i
    STORE I         // Store the new value of i

    LOAD FIVE       // Load the loop end condition (5)
    SUBT I          // Subtract i from 5
```

AC FFFF
IR 7000
MAR 10F
MBR 7000
PC 110
IN 0000
OUT 0002

OUTPUT MODE: DEC

```
0
1
2
```

Machine halted normally.

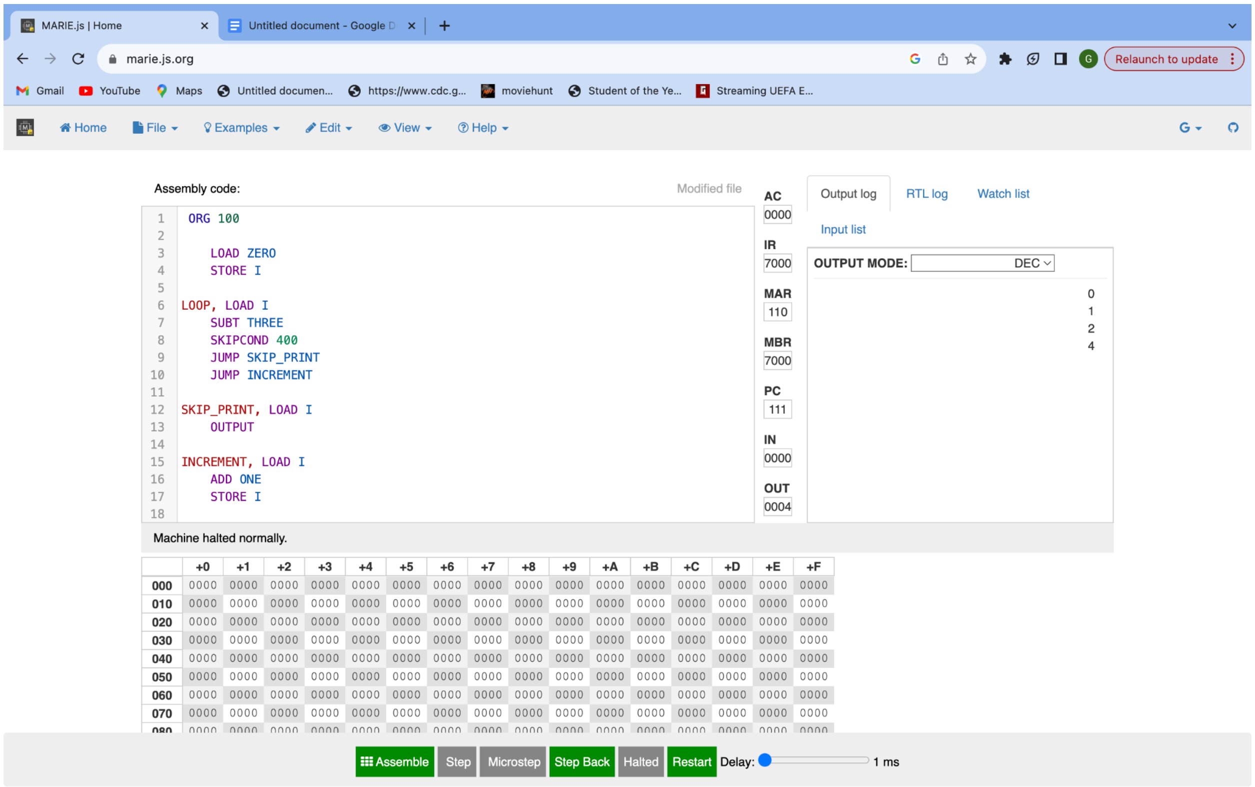
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
020	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
030	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
040	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
050	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
060	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
070	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
080	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

Delay: 1 ms

Qno.3)

```
    ORG 100

    LOAD ZERO
    STORE I

LOOP, LOAD I
    SUBT THREE
    SKIPCOND 400
    JUMP SKIP_PRINT
    JUMP INCREMENT

SKIP_PRINT, LOAD I
    OUTPUT

INCREMENT, LOAD I
    ADD ONE
    STORE I

    LOAD FIVE
    SUBT I
    SKIPCOND 400
    JUMP LOOP

END_LOOP, HALT

I, HEX 0
ONE, DEC 1
THREE, DEC 3
FIVE, DEC 5
ZERO, HEX 0
```

Output:



```
ORG 100

    LOAD ZERO
    STORE I

LOOP, LOAD I
    SUBT THREE
    SKIPCOND 400
    JUMP SKIP_PRINT
    JUMP INCREMENT

SKIP_PRINT, LOAD I
    OUTPUT

INCREMENT, LOAD I
    ADD ONE
    STORE I

```

AC 0000
IR 7000
MAR 110
MBR 7000
PC 111
IN 0000
OUT 0004

OUTPUT MODE: DEC

```
0
1
2
4
```

Machine halted normally.

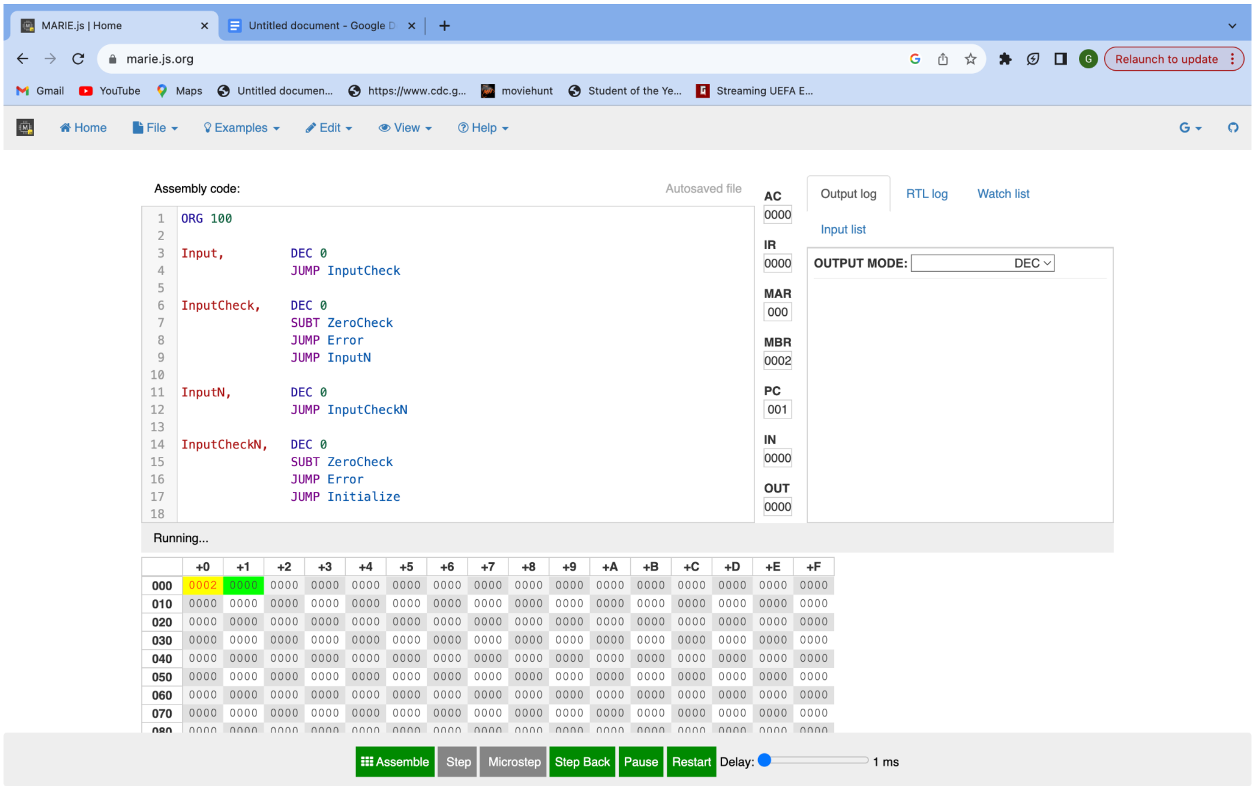
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
020	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
030	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
040	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
050	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
060	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
070	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
080	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

Qno.4)

```
ORG 100

Input,        DEC 0
            JUMP InputCheck

InputCheck,    DEC 0
            SUBT ZeroCheck
            JUMP Error
            JUMP InputN

InputN,       DEC 0
            JUMP InputCheckN
```

```
InputCheckN,   DEC 0
               SUBT ZeroCheck
               JUMP Error
               JUMP Initialize

Error,         JUMP Halt

ErrorMsg,      DEC -1

Initialize,    STORE M
               STORE N
               CLEAR
               JUMP Loop

Loop,          ADD M
               SUBT One
               SUBT ZeroCheck
               JUMP Loop

Halt,          HALT

One,           DEC 1

M,             DEC 0
N,             DEC 0

ZeroCheck,     DEC 0

               END Start
```

Output:



Assembly code:

```
ORG 100

Input,          DEC 0
                JUMP InputCheck

InputCheck,     DEC 0
                SUBT ZeroCheck
                JUMP Error
                JUMP InputN

InputN,         DEC 0
                JUMP InputCheckN

InputCheckN,    DEC 0
                SUBT ZeroCheck
                JUMP Error
                JUMP Initialize

```

AC 0000

IR 0000

MAR 000

MBR 0002

PC 001

IN 0000

OUT 0000

OUTPUT MODE: DEC

Running...

	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
000	0002	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
010	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
020	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
030	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
040	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
050	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
060	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
070	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
080	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

Delay: 1 ms