

1) Write a MARIE assembly language to read two numbers and determine which is smaller.

HLL:

```
read X
read Y
if X < Y then
    min = X
else
    min = Y
end if
output min
```

2. Consider the microoperations of the fetch-decode-execute machine cycle in the execution of the “Simple Program” below that calculates $RESULT = X + Y$.

<u>Address</u>	<u>Label</u>	<u>Assembly Language</u>	<u>Machine Language</u>
100		LOAD X	1104 ₁₆
101		ADD Y	3105 ₁₆
102		STORE RESULT	2106 ₁₆
103		HALT	7000 ₁₆
104	X,	DEC 35	0023 ₁₆
105	Y,	DEC -23	FFE9 ₁₆
106	RESULT,	DEC 0	0000 ₁₆

Revised Figure 4.14 (a) LOAD X (1104₁₆ in ML)

Step	Step #	RTN	PC	IR	MAR	MBR	AC
(initial values)			100				
Fetch	T ₀	MAR ← PC	100		100		
	T ₁	MBR ← M[MAR]	100		100	1104	
	T ₂	IR ← MBR	100	1104	100	1104	
	T ₃	PC ← PC + 1	101	1104	100	1104	
Decode IR[15-12]	T ₄	MAR ← IR[11-0]	101	1104	104	1104	
Get operand	T ₅	MBR ← M[MAR]	101	1104	104	0023	
Execute	T ₆	AC ← MBR	101	1104	104	0023	0023

Revised Figure 4.14 (b) ADD Y (3105₁₆ in ML)

Step	Step #	RTN	PC	IR	MAR	MBR	AC
(initial values AFTER LOAD 104)			101	1104	104	0023	0023
Fetch	T ₀	MAR ← PC	101	1104	101	0023	0023
	T ₁	MBR ← M[MAR]	101	1104	101	3105	0023
	T ₂	IR ← MBR	101	3105	101	3105	0023
	T ₃	PC ← PC + 1	102	3105	101	3105	0023
Decode IR[15-12]	T ₄	MAR ← IR[11-0]	102	3105	105	3105	0023
Get operand	T ₅	MBR ← M[MAR]	102	3105	105	FFE9	0023
Execute	T ₆	AC ← AC + MBR	102	3105	105	FFE9	000C

Revised Figure 4.14 (c) STORE RESULT (2106₁₆ in ML)

Step	Step #	RTN	PC	IR	MAR	MBR	AC
(initial values AFTER LOAD 104)			102	3105	105	FFE9	000C
Fetch	T ₀						
	T ₁						
	T ₂						
	T ₃						
Decode IR[15-12]	T ₄						
Execute*	T ₅						
	T ₆						

* “Get Operand” step is not necessary for STORE instructions