

Name: _____ **Date:** _____ **Lab Section:** _____

Objectives: To review important concepts in Chapter 4 - MARIE. Answer on this sheet where space is given.

References: ECOA2e Section 4.1-4.6, 4.8.1-4.9.1, 4.9.3, 4.10, 4.11.1, 5.4.2 and associated Chapter Slides
 Class Notes (via course home page): **text_errata.txt**

Equipment: MARIE Simulator: free download from **http://computerscience.jbpub.com/ecoa/2e**
 and the Sun Java Run Time Environment

Not all questions will be marked – check all your answers against the answer sheet when it is posted.

1. Open MARIE and do the following:

Label	Mnemonic
	load x
	add y
	store z
	halt
x,	dec 32
y,	dec -15
z,	hex 0

Type in the adjacent four-instruction MARIE assembly language program, save it, assemble it to object code, load it into MARIE, and prepare to single-step through it.

Single-step through each instruction of the program and write the contents of AC, IR, MAR, MBR and PC in the following table after each instruction has been fetched and executed.

Be prepared to explain why the registers contain these values.

Remember that the PC always points to the *next* instruction when executing the *current* instruction.

The cycle is always: *fetch, increment, execute*

Instruction	AC	IR	MAR	MBR	PC
load x					
add y					
store z					
halt					

2. Hand-assemble (without using MARIE) and answer on separate paper Question 13 on page 239.
3. Answer on separate paper Question 14 on page 239.
4. Question 15a p.239: i) _____ ii) _____ iii) _____
5. Read *Address Modes*, Section 5.4.2, and refer to Figure 5.3, Table 5.1, and Question 13 p.277. Given the instruction **LOAD 2000**, determine the actual value loaded into the accumulator and fill in the Addressing Move table below if the index register **R1** contains the value **1200**:

<i>Memory</i>	
Address	Contents
1200	2500
...	...
2000	2200
...	...
2200	2600
...	...
2800	1200
...	...
3200	3600
...	...
3600	2400

<i>Addressing Mode</i>	<i>Value loaded in accumulator, AC</i>
Immediate	
Direct	
Indirect	
Indexed (with R1)	

Note that most MARIE instructions are Direct Addressing. MARIE has no Immediate or Indexed Addressing instructions and only two Indirect Addressing instructions: **AddI** and **JumpI**. (MARIE has no **SubI**, **LoadI**, or **StoreI** – a real computer ISA would be more consistent and permit more address modes.)

6. Give the RTL/RTN for a new MARIE **LoadI** instruction:

7. On separate paper, fill in the four-element addressing mode table for Question 13, p.277.
8. On separate paper, fill in the four-element addressing mode table for Question 14, p.277.