

Name: Sang Hwa Lee

HW01: Representing Data and Instructions

Homework assignments are to be complete *individually*. Asking another human for an answer (e.g., posting on forums, chat, or messaging) is considered academic dishonesty. Please see the syllabus for further guidance about what it means to complete an assignment individually. This assignment is worth a total of **25 points** and the value in parenthesis indicates the point value for that question.

SECTION 00: NUMBER SYSTEMS

1. (1pt) In the binary number 01000110 what place value is represented by the underlined digit?

64

2. (1pt) What does the “0x” tell us about the number 0x010C?

The prefix 0x is used in code to indicate that the number is being written in hex(16).

3. (1pt) How many bits are represented by 3 digits of hexadecimal (e.g., 0xFFF)?

12

4. (1pt) How many bytes are represented by 4 digits of hexadecimal (e.g., 0xFFFF)?

2

5. (1pt) In the number 0x50AC what place value is represented by the digit “A”?

1010

6. (1pt) Convert 129 into binary.

10000001

7. (1pt) Convert 129 into hexadecimal (hint: look at each nibble above).

81

8. (1pt) Is the binary number 10011000 odd or even?

Even

9. (1pt) Convert 0x1D to decimal.

29

10. (1pt) Convert the binary number 11100001 to hexadecimal (hint: look at each nibble).

E1

CONTINUES ON NEXT PAGE

SECTION 01: INSTRUCTIONS

The following section will be asking questions about the assembly program in the following table. The Instruction Set is defined in the table on the last page of this assignment. The *Address* column represents the memory address where the machine code instructions that the Assembler produces will be loaded into when the program is running.

Address	Assembly Language
00	INP
01	STA 16
02	INP
03	SUB 15
04	STA 17
05	LDA 18
06	ADD 16
07	STA 18
08	LDA 17
09	SUB 15
10	STA 17
11	BRP 04
12	LDA 18
13	OUT
14	HLT
15	DAT 01
16	DAT 00
17	DAT 00
18	DAT 00

11. (1pts) What is the machine code for the instruction in memory address 00?

901

12. (1pts) What is the mnemonic for the instruction in memory address 01?

STA

13. (1pts) What is the opcode for the instruction in memory address 01?

3

14. (1pts) What is the operand for the instruction in memory address 01?

16

15. (1pts) During the execute phase of the instruction at memory address 00 the user input the value "9". Explain what the instruction in memory address 01 does during the execute phase.

Store the contents(9 – value) in the accumulator in the memory address(16 – address number)

CONTINUES ON NEXT PAGE

16. (1pts) During the execute phase of the instruction at memory address 02 the user input the value "2". Explain what the instruction in memory address 03 does during the execute phase.

Subtract the value stored in the memory address 15 from the accumulator

17. (3pts) The Instruction Counter is currently "11" and the Accumulator contains the value "1". Explain what will happen during the Fetch, Decode, and Execute phases of this one instruction cycle.

1. First, when we think of the fetch cycle, the program brings the value of instruction register(3) and address register(17) stored in RAM10 to CPU.
2. Fetch cycle is complete, start decode instruction.
The 11th task is STA17, which is to store the current value at address 17.
3. Finally, Execute step. Because the value remaining in the accumulator is 1, the value of 1 will be stored at address 17.

Use the LMC simulator at <http://peterhigginson.co.uk/LMC/> to observe this program running by copying the contents of the Assembly Language column into the LMC. Make sure that you click on the RESET button before re-running the program or you will not get the correct results.

18. (1pts) What is the output if you use the inputs "2" and "1"?

2

19. (1pts) What is the output if you use the inputs “2” and “2”?

6

20. (1pts) What is the output if you use the inputs “2” and “3”?

12

21. (1pts) Based on your observations, what basic mathematical operation is this program performing on the two inputs?

The formula for the general term of the sequence $\Rightarrow a_n = n^2 + n$

Ex) Q)18. $a_1 = 1^2 + 1 = 2$

Q)19 $a_2 = 2^2 + 2 = 6$

Q)20 $a_3 = 3^2 + 3 = 12$

CONTINUES ON NEXT PAGE

The LMC has a very reduced instruction set. The program you observed is a long way to implement a basic mathematical operation when there was no instruction defined for it in the LMC instruction set. Pretend that you are the engineer for the next version of the LMC CPU and define the new LMC instruction for opcode 4 that would make the observed program easier to write.

22. Fill in the blank cells to define the new instruction here: (1pt)

Instruction	Mnemonic	Opcode	Operand	Description
Division	DIV	4	xx	Divide the contents of the memory address to the Accumulator

23. Fill in the blank cell to use the new instruction to create a much shorter version of the program you observed running: (1pt)

Memory Address	Machine Code	Assembly Language
00	901	INP
01	306	STA 06
02	901	INP
03	406	DIV 06
04	902	OUT
05	000	HLT
06	000	DAT 00

INSTRUCTION SET

This is the instruction set for the Little Man Computer by Dr. Stuart Madnick ("Little Man Computer.").
The opcode is 1 decimal digit; the operand is 2 decimal digits.

Note: There is no instruction defined for the opcode 4.

Instruction	Mnemonic	Opcode	Operand	Description
Halt	HLT	0	00	Stop running
Add	ADD	1	XX *	Add the contents of the memory address to the Accumulator
Subtract	SUB	2	XX *	Subtract the contents of the memory address from the Accumulator
Store	STA	3	XX *	Store the contents in the Accumulator in the memory address
Load	LDA	5	XX *	Load Accumulator with contents of memory address
Branch always	BRA	6	XX *	Set the PC to the value in the operand
Branch if zero	BRZ	7	XX *	Set the PC to the operand if the Accumulator is zero
Branch if zero or positive	BRP	8	XX *	Set the PC to the operand if the contents of the Accumulator is zero or positive
Input	INP	9	01	Retrieve user input and store in the Accumulator
Output	OUT	9	02	Output the contents of the Accumulator
Data storage	DAT			Label for a memory address; also can have a contents specified

* XX refers to a memory address.

"Little Man Computer." In *Wikipedia*, January 27, 2020.

https://en.wikipedia.org/w/index.php?title=Little_man_computer&oldid=937810035.