Assignment Instructions

Below are the questions for the assignment. Write your answers in the separate Answer document Optional challenge questions are just for your own learning; they do not count toward grading.

For ALL Word processing documents, you must submit your documents in one of the following formats: MS-Word (.doc/.docx) or PDF (common read only format). They will be returned ungraded if submitted in any other format.

Submit assignments all assignments via elearn - attach your word document or PDF to your submission. Do NOT copy and paste your document into the submission text box.

Machine Architecture

Q1: Give a one-two sentence description of each of these components:

a) ALU:
b) Control Unit:
c) ACC (accumulator):
d) PC (program counter):

Q2: A computer's main and secondary memory both store information. Why are there two different "memories"?

Q3: For each step of the machine cycle, give a sentence or two description of what happens:

a) Fetch: What is fetched? From where to where?
b) Decode: What information is "decoded"?
c) Execute: What components might be involved in execution?

Little Man Computer

Q4:

a)Translate this machine code to English descriptions:

Address	Value	
00	901	
01	310	
02	901	
03	110	
04	310	
05	110	
06	310	
07	110	
08	902	
09	000	

b) Call the inputs x, y. **Describe** the output of the program. (What expression in terms of x and y does it calculate?) *DO NOT list what it does step by step.*

Q5: Write LMC machine code for a program that gets two numbers from input (call the first one **x** and second one **y**) and outputs the result of calculating **y - 2x**. Given inputs of 3 and 8, the output should be 2.

Hint: you are going to have to store the inputs away in memory so they do not wipe each other out…
Partial credit available, make sure to submit something, even if you think it has a bug.

Error Detection & Correction:

Q6: What is a major flaw of the repetition trick as compared to the redundancy trick?

Q7: Using the encoding chart shown below for a (7, 4) Hamming code, you get the message 0000111. How should you interpret the message? Why?

0000 → 0000000	1000 → 1000110
0001 → 0001011	1001 → 1001101
0010 → 0010111	1010 → 1010001
0011 → 0011100	1011 → 1011010
0100 → 0100101	1100 → 1100011
0101 → 0101110	1101 → 1101000
0110 → 0110010	1110 → 1110100
0111 → 0111001	1111 → 1111111

Q8: Say we have the message 3 9 2 7

a) Give the checksum we would get using the last digit of sum checksum (show work):
b) Give a sequence with two errors that produces the same checksum (show work):
c) Give the staircase checksum for the original message (show work):
d) Give the staircase checksum for the erroneous sequence from part b (show work):

Q9: The table below shows a 9-digit message along with the pinpoint checksums:

3	4	5	2
3	2	6	4
8	3	7	8
7	9	8	

a) How can you tell where the error is?

b) What should the correct message be? (Fix the message shown below.)
3 4 5
3 2 6
8 3 7

Programming:

Complete Code.org lessons assigned this week. This will go into your "participation grade". There is nothing to submit for this – I will check your progress through Code.org. Partial credit is possible if you only finish some of the parts.

You will get NO CREDIT FOR THIS if your display name in Code.org is not set to your Chemeketa user name.
(for bsmith12@my.chemeketa.edu the display name should be **bsmith**)

Assignment Problems:

These are a graded part of this assignment. You will paste a screenshot of your drawing into the answer document along with the text for your code (NOT a screenshot of the code).

Q10:

Give the final value for x after the code has run in each of these samples: (Hint, do them as shown in 5.4.18)

a)	var a = 2; var x = a + 3;

b)	var x = 5;
	var y = 2;
	y = x;
	x = y;

c)	var x = 8; x = x + 2;

d)	var a = 2;
	var b = 4;
	var c = 5;
	c = a + b;
	a = 7;
	var x = c * b;

Q11:

Use the starter code in 5.5.21 to make your own clicker game.

This is a multi-level problem. Level 4 is full credit, lower levels are partial. You are better off submitting something that is correct from a lower level than a broken version of a higher level.

Submit a screenshot of the game_screen and the text version of your code.

- Level 1: You have customized the graphics and text of the game.
- Level 2: You make a variable for scores and lives, update the variables in the game and update the display on the screen with the current values.
- Level 3: The game will switch to the win or loss screen when points and lives reach some value (you can pick the score that is the goal).
- Level 4: Restarting the game resets all the counters so the game plays correctly again.

Feel free to add extra features on top of these basics.