

Recitation Assignment #1

Please do the following problems during your recitation session, including any additional problems given to you by your TA. Within 72hrs of your recitation session, you must upload the complete solutions to these problems to Sakai, so that your TAs can evaluate them in a timely fashion. **For this recitation, please show an instructor or learning assistant that you have completed each assignment, and verify that they have given you credit. If you can't complete the assignments before the end of recitation, or don't get verification, please submit either a PDF or word document with your written solutions, and then any .m files with code. You can type up your written solutions, use a tablet to write them down, or take a VERY CLEAR picture of them and put that picture in a document.** Collaborative problems can be worked on in teams of up to 5 people, as long as each team member individually completes the problem uploads the solution as part of their own Sakai submission or shows the solution to their instructor individually, and lists the names of all collaborators in the Sakai submission. Collaboration and discussion of solutions is not permitted for questions labeled as individual problems.

Prior to working on these problems, open MATLAB. Then create a variable with any value you want in the command line. Then find that variable in the workspace. If you have trouble with any of this, let an instructor or learning assistant know and they will help you.

1. [Collaboration] Recall the formula for how many different values can be stored in x bits. Calculate how many different values can be stored in 5 bits. Show your work (i.e., write the formula, fill in the value, and show your answer).

Let's say that we use those 5 bits to store integer values. 00000 represents integer 0, 00001 represents integer 1, 00010 represents integer 2, etc. Create a table that has two columns. In the right column, list all of the integers that can be represented by 5 bits. In the left column, write out the bits in the format shown above.

Now let's say that we want our integer to be either *positive* or *negative*. We can use one of our bits to signify whether a number is positive or negative. Let's say that 00000 is still 0, and 00001 is still 1, but 11111 is now "-0" and 11110 is now "-1," etc. This format is called the *one's compliment*. Create a new rightmost column in your table, writing all of the signed integer values to the corresponding combination of bits. What does this format mean for the number of different values that can be stored? What does this format mean for the *magnitude* of the highest (or lowest) number you can store?

Let's try one more format. 00000 is still 0, 00001 is still 1. However, -1 is now 11111, and 11110 is now -2. This format is called the *two's compliment*. Create a new rightmost column in your table, writing all of the signed integer values that correspond with all of the combinations of 4 bits. What does this format mean for the number of different values that can be stored? What does this format mean for the *magnitude* of the highest (or lowest) number you can store?

2. [Collaboration] The first electronic word processor, the Wang 1200, could handle one line of text at a time, which could be edited before printing or storing on as data on a cassette tape. Each line of text was stored in 100 Bytes, and could be a maximum of 100 characters long. How many bits per character is that? How many possible symbols could be represented in this way? How many bits would it take to represent all of the lowercase letters in the Latin alphabet used for English? What about upper and lower case? What about all of the letters, numbers, and punctuation on

your computer's keyboard? Space, Enter and Tab can all count as one symbol. Don't forget to show your work. Your answers might be different from your work partners, as some keyboards may have a few different keys.

3. [Collaboration] Create a script in MATLAB. Create a value and give it the value "1/3". Then multiply that variable by 3, storing the value in the same variable (i.e. $x = x*3$). Comment your code. After you run this code, look in the workspace and find your variable. Double click on it. When you do this, you should get a new tile in MATLAB. Find the number in the grid and double click on that. What happened? Write a comment in your code describing this.
4. [Collaboration] Do the following arithmetic problems in binary. If a number goes higher than your leftmost bit, create a *new* bit:

5 + 5
8 + 7
9 - 3
7 - 5

Please show all of your work (carried numbers, conversion back to base 10).

5. [Collaboration] There is a system for converting binary numbers to decimal numbers. Starting with the rightmost digit, and counting from zero, multiply the value of the bit (0 or 1) by 2 raised to the number you are counting. Then sum all of those numbers together. For example, 10010011 is:

$$2^7 * 1 + 2^6 * 0 + 2^5 * 0 + 2^4 * 1 + 2^3 * 0 + 2^2 * 0 + 2^1 * 1 + 2^0 * 1$$

You don't have to write out the numbers you are multiplying by zero bits, as they have no value. You can also simplify what you write out by just writing the powers of 2, like this:

$$2^7 + 2^4 + 2^1 + 2^0 = 128 + 16 + 2 + 1 = 147$$

Try it with the following numbers:

1 0 1 1

1 0 1 0 0 0 0

1 1 0 0 1 1 1 1

0 1 1 1 1 0 0 0

1 0 1 0 0 0 0 0 1 1 1 0 1 0 0 0