# LBSC 690: Information Technology
# Homework 01:
# Computers and Data

William Webber
CIS, University of Maryland

Spring semester, 2012

Answers due via email to `wew@umd.edu` by Tuesday, 7th February, at 3:30pm ET.

## Question 1: Binary addition

Adding two multiple-digit binary numbers together is similar to adding two "regular" (decimal) numbers together. We work a column at a time, right to left, add each column, and carrying any remainder over into the next column to the left. Binary addition, however, takes a particularly simple form, because only two values are involved.

**Part a**:

Add the following (6-bit) binary numbers together (please do the working carefully, but you only need to show me the answers):

$$001010 \quad + \quad 011001 = \tag{1}$$
$$010111 \quad + \quad 010110 = \tag{2}$$
$$011111 \quad + \quad 000001 = \tag{3}$$

**Part b**:

Convert the above (operands and results) into decimal, and check that the addition is correct.

□

Binary arithmetic in modern computers is done to a fixed width. That is, if you add two 6-bit values together, then the result also have to fit into 6 bits. Any carry over from the left-most bit is thrown away.

**Part c**:

Obeying the rules of fixed-width binary arithmetic, and assuming that we are working with 6-bit values, compute the following sum:

$$111111 \quad + \quad 000001 = \tag{4}$$

□

Under the original UNIX operating system, the current time is represented as a $32$-bit number, counting the number of seconds since January 1st, 1970. As each second ticks by, one is added to this internal time stamp.

**Part d**:

What is the last date (in seconds, and in years — approximation is ok) that the $32$-bit UNIX timestamp is able to represent correctly?

**Part e**:

What time will the UNIX timestamp show when we are one second past the last date it can represent?

[[ Note: this limitation has been fixed in most modern UNIX systems by moving to a $64$-bit timestamp—you might want to think about when this will run out. But there may be some historical UNIX system that will encounter this problem when the fatal time comes. ]]

# Question 2: Sneaker-net

When we think of transferring data from one computer to another nowadays, we almost automatically think of doing so using a network. An alternative, though, is to copy the data to some (semi-) removable medium, and then physically carry that medium to the destination. Such a method is facetiously known as "sneaker-net", since we are not using cables to communicate the data (as in ethernet and similar network techonologies), but our sneakers.

Assume that I need to transfer $100$ GB of data from a source computer to a destination computer, and that I have two possibilities for transferring the data:

1. Send it over a network, which has a speed of $100$ Mbps; or
2. Unplug the hard drive the data is on, carry it to the other computer (walking at $5$ feet per second), and plug it in.

Assume that unplugging and plugging-in the hard drive take no time; the only elapsed time is in walking.

**Part a**:

Assuming I have to walk $1{,}000$ feet to the destination computer, what is the effective bandwidth of our sneaker-net connection?

**Part b**:

How far away does the destination computer have to be before it is faster to send the data over the network than to carry it by sneaker-net?