COSC 065 - Hardware Systems
Homework #3 - Representation
Due: 2006 September 28, beginning of lecture
This assignment is to be completed individually.

1. In lecture we looked at the range of 8-bit, two's complement numbers. In table format, give the range of 4-digit numbers in radix complement with radix equal to 2, 10, and 16. For each radix, include the two most positive values, the values at and near zero, and the two most negative values. The actual range (the meanings of these representations) should be clearly indicated in our native base-10 number system (with minus sign for negative values).

2. The textbook's discussion of two's complement numbers is followed by a section on hardware sign extension. Briefly, describe why most modern computer hardware performs sign extension. Give a concrete example where this is desirable; give a concrete example where this will do the wrong thing.

3. In examining the IEEE Floating Point standard, we neglected to mention a couple of special cases related to the *exponent* (e') field. Recall that the exponent is stored in *excess-B* format, with a *bias* of 127. Thus, when the true exponent of the floating point value is $e = 0$, the stored exponent field contains $e' = 127$. However, two possible values of e' have special meaning: when $e' = 0$, rather than mean an exponent of $e = -127$, it means that $e = -126$ and the *mantissa* is NOT normalized. On the other end of the spectrum, when $e' = 255$, this does not mean $e = 128$, but rather is used to represent infinity and some special error conditions.

   Given this new information, what are the largest and smallest (magnitude) possible values that can be represented in IEEE "Single Precision" Floating Point format? Include both the bit representations and the base-10 equivalent values in your answer.

4. Consider the equation $4x^2 - 50x + 136 = 0$. If solutions are $x = 4$ and $x = 4\frac{3}{4}$, what numeric base am I working in?

5. A 32-bit register contains the hexadecimal value 0x44722E44. What is the meaning of this value in:

   - unsigned binary format?
   - two's-complement format?
   - IEEE single precision floating point format?
   - ASCII format?

   Repeat this question with the hex value 0x80485732.

6. Lab question: Give a schematic for a *full adder*. Clearly label your three inputs (X, Y, and C_{in}) and your two outputs (S, and C_{out}). Include pin numbers for your gates, and LED circuits for displaying all five bits. The full adder can be built from two half adders, or it can be built directly. You may use the full range of gates you have already built out of NANDs: AND, NAND, OR, NOR, NOT, and XOR.