# Homework 1

**Homework 1 is due on Wednesday, February 26, at the start of the lecture. Remember to include your *Discussions section* (e.g. E1) and follow the complete Homework submission guidelines.**

**Please ask all questions about this assignment during the office hours, or email your instructor.**

Be sure to show your work on all your homework problems; otherwise, you risk not receiving credit.

It is recommended that you do not use a calculator (or online tools, etc.) when you do your homework. Remember that there will be no calculators allowed on exams.

## Abstraction and Binary Representation

### 1. Abstraction

Do problem 1.7 from Patt and Patel. Limit your answer to at most 3 sentences.

### 2. Properties of the algorithm

Consider the following instruction: "Go straight for a mile or so and then turn left." What property of an algorithm this instruction does *not* have that makes it unacceptable as a statement in an algorithm?  Explain your answer in no more than 3 sentences.

### 3. Required number of bits

In Fall 2016, there were 33467 undergraduate students enrolled at University of Illinois at Urbana-Champaign. Answer the following questions and show your work for full credit.

1. If every undergraduate student is to be assigned a unique bit pattern, what is the minimum number of bits required to do this?
2. How many more undergraduate students can be enrolled without requiring additional bits for each student's unique bit pattern?

### 4. Conversion to 2's complement

Convert the following decimal numbers into 8-bit 2's complement representation. Show your work for full credit.

1. 12
2. -25

### 5. Limits of representation

1. How many numbers can be represented using 7-bit 2's complement representation?
2. What is the smallest negative and the largest positive numbers that can be represented with 7 bits in 2's complement representation? Convert these numbers from decimal to 2's complement binary representation, show your work.

### 6. Conversion from 2's complement

Convert the following 8-bit 2's complement numbers into their decimal representation. Show your work for full credit.

1. 01011010
2. 10110110
3. 01100111
4. 11000011