Homework 6
Due October 24, 2013

CMSC 203

1. Section 5.1, Exercise 4
   Let $P(n)$ be the statement that $1^3 + 2^3 + \ldots + n^3 = (n(n+1)/2)^2$ for the positive integer n.

   a) What is the statement $P(1)$?
      $1^3 = (1(1+1)/2)^2$

   b) Show that $P(1)$ is true, completing the basis step of the proof.
      $1^3 = (1(1+1)/2)^2 \rightarrow 1 = (2/2)^2 \rightarrow 1 = 1$

   c) What is the inductive hypothesis?
      $P(k)$ is true, i.e. $1^3 + 2^3 + \cdots + k^3 = (k(k+1)/2)^2$ for some $k \in \mathbb{N}$.

   d) What do you need to prove in the inductive step?
      That $P(k+1)$ is true, i.e. that $1^3 + 2^3 + \cdots + k^3 + (k+1)^3 = ((k+1)(k+1+1)/2)^2$.

   e) Complete the inductive step, identifying where you use the inductive hypothesis.

$$1^3 + 2^3 + \cdots + k^3 + (k+1)^3 = \sum_{i=1}^{k+1} i^3$$
$$= (k+1)^3 + \sum_{i=1}^{k} i^3$$

apply the inductive hypothesis...

$$\begin{aligned}
&= (k+1)^3 + (k(k+1)/2)^2 \\
&= (k+1)^3 + ((k^2+k)/2)^2 \\
&= 4(k^3+3k^2+3k+1)/4 + (k^4+2k^3+k^2)/4 \\
&= (k^4+6k^3+13k^2+12k+4)/4 \\
&= ((k+1)(k+2)/2)^2
\end{aligned}$$

   f) Explain why these steps show that this formula is true whenever n is a positive integer.
      Any interpretation of the Principle of Mathematical Induction is acceptable

2. Section 5.1, Exercise 20
   Prove that $3^n < n!$ if n is an integer greater than 6.
   Basis step:
   $P(7) = 3^7 = 2187 < 5040 = 7!$
   Inductive step:
   Hypothesis: $P(k) = 3^k < k!$
   Prove: $P(k+1) \rightarrow 3^{k+1} < (k+1)!$

$$
\begin{aligned}
3^{(k+1)} &= 3 \cdot 3^k \\
&< 3 \cdot k! \\
&< (k+1) \cdot k! \text{ [when } k > 6] \\
&= (k+1)!
\end{aligned}
$$

3. Section 5.1, Exercise 50
   What is wrong with this "proof"?
   "*Theorem*" For every positive integer n, $\sum_{i=1}^{n} i = (n + \frac{1}{2})^2/2$

   Basis Step: The formula is true for $n = 1$.

   Inductive Step: Suppose that $\sum_{i=1}^{n} i = (n+\frac{1}{2})^2/2$. Then $\sum_{i=1}^{n+1} i = (\sum_{i=1}^{n} i)+(n+1)$. By the inductive hypothesis, $\sum_{i=1}^{n+1} i = (n + \frac{1}{2})^2/2 + n + 1 = (n^2 + n + \frac{1}{4})/2 + n + 1 = (n^2 + 3n + \frac{9}{4})/2 = (n + \frac{3}{2})^2/2 = [(n + 1) + \frac{1}{2}]^2/2$, completing the inductive step.
   The basis step is not actually proved, likely because it is not actually true: $P(1)$ is false.

4. Section 5.2, Exercise 12
   Use strong induction to show that every positive integer n can be written as a sum of distinct powers of two, that is, as a sum of a subset of integers $2^0 = 1, 2^1 = 2, 2^2 = 4$, and so on. [*Hint*: For the inductive step, separately consider the case where $k + 1$ is even and where it is odd. When it is even, note that $(k + 1)/2$ is an integer]
   Basis step:
   $P(1) = 1$ can be written as 2^0
   Inductive step:
   Hypothesis: $P(j) = j$ can be written as the sum of distinct powers of 2, for all $j \leq k$
   Prove: $P(k + 1) = k + 1$ can be written as the sum of distinct powers of 2

   Case 1: $k + 1$ is even
   If $k + 1$ is even, then $k + 1 = 2x$ for some $x <= k$
   By the inductive hypothesis, $x = a_1 2^0 + a_2 2^1 + \cdots + a_n 2^n$ for coefficients $a_i \in \{0, 1\}$
   Then $k + 1 = 2(a_1 2^0 + a_2 2^1 + \cdots + a_n 2^n)$
   $= a_1 2^1 + a_2 2^2 + \cdots + a_n 2^{n+1}$
   So $k + 1$ can be written as the sum of distinct powers of two

   Case 2: $k + 1$ is odd
   By our previous logic, $k + 1$ can be written as $1 + a_1 2^1 + a_2 2^2 + \cdots + a_n 2^{n+1}$.
   Since $1 = 2^0$ and our expansion of powers of two doesn't already include 2^0,
   $k + 1 = 2^0 + a_1 2^1 + a_2 2^2 + \cdots + a_n 2^{n+1}$

5. Section 5.2, Exercise 32
   Find the flaw with the following "proof" that every postage of three cents or more can be formed using just three-cent and four-cent stamps.

   Basis Step: We can form postage of three cents with a single three-cent stamp and we can form postage of four cents using a single four-cent stamp.

   Inductive Step: Assume that we can form postage of j cents for all nonnegative integers j with $j \leq k$ using just three-cent and four-cent stamps. We can then form postage of $k + 1$ cents by replacing one three-cent stamp with a four-cent stamp or by replacing two four-cent stamps by three three-cent stamps.

   Saying that the basis step is not strong enough might be partially true, but it generalizes a pattern we saw with other postage problems without demonstrating why the basis case needs to be large. In this case, the theorem is untrue, so we could make the basis case as strong as we liked, we still wouldn't be able to form postage of 5 cents, for example. In this case, the main problem is that the inductive step does not give us a 100% method of generating $k+1$ postage. Particularly, it only works when the method for making k postage involved either a 3 cent stamp or two 4 cent stamps, without giving any guarantee/reasoning for why there must be such a case in the postage for k. As it turns out, this causes the theorem to fail for small values of k.

6. Section 5.4, Exercise 10
   Give a recursive algorithm for finding the maximum of a finite set of integers, making use of the fact that the maximum of n integers is the larger of the last integer in the list and the maximum of the first $n-1$ integers in the list.

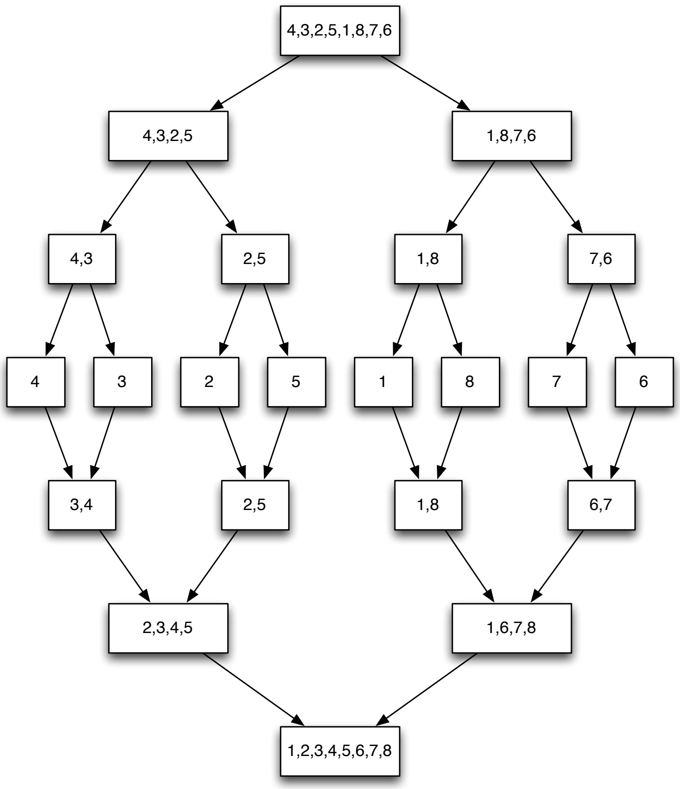
   The algorithm needs to have a base case and some kind of recursive divide+merge, though they can be combined into one step

   ```
   function LISTMAXIMUM(list)
       if size(list) = 1 then                          ▷ base case
           return list[0]
       end if
       m = LISTMAXIMUM(list[0..n − 1])                 ▷ divide step
       return MAX(m, list[n])                          ▷ merge step/conquer step
   end function
   ```

7. Section 5.4, Exercise 22
   Prove that the recursive algorithm you found above is correct.

   Just about any inductive proof will probably work here Basis step:
   $P(1) =$ `LISTMAXIMUM` returns the correct value for a list with only one element. This is covered by the base case of the algorithm.
   Inductive step:
   Hypothesis: Assume $P(k)$, i.e. that `LISTMAXIMUM` returns the correct value for any list with k elements.
   Prove: $P(k+1)$, i.e. that `LISTMAXIMUM` returns the correct value for any list with $k+1$ elements.
   `LISTMAXIMUM` solves a list of $k+1$ elements by returning the maximum of a sublist with k of those elements plus one more. By our hypothesis, the maximum of that sublist will be correctly calculated by the algorithm, so we have the true max of that sublist and the one missing element. We can construct a trivial proof by cases which demonstrates that returning the max of those two will be the true max of that list. *This is a hand-wavey proof, but good enough for me.*

8. Section 5.4, Exercise 44
   Use a merge sort to sort $4, 3, 2, 5, 1, 8, 7, 6$ into increasing order. Show all steps used by the algorithm.



9. Section 5.4, Exercise 50
   Sort $3, 5, 7, 8, 1, 9, 2, 4, 6$ using the quick sort.