CS311H Problem Set 7

Due Tuesday, November 5

1. (10 points) Prove by induction that the sum of the first n positive odd integers is n^2. Explicitly state whether you are using regular or strong induction.

2. (10 points) Use induction to prove that the following equality holds for every positive integer n:

$$\sum_{i=1}^{n} i \cdot 2^i = (n-1) \cdot 2^{n+1} + 2$$

   State whether you are using regular or strong induction.

3. (5 points) Explain what is wrong with the following "proof":

   Claim: $\forall n \geq 0.\ 3n = 0$
   Proof: By strong induction on n. For the base case, we have $n = 0$. Since $3 \cdot 0 = 0$, the claim holds for the base case.
   For the inductive step, we prove the property for $k+1$, i.e., $3 \cdot (k+1) = 0$. Observe that $k+1$ can be written as $i+j$ for some natural numbers i, j where $i \leq k$ and $j \leq k$. Now, by the inductive hypothesis, we have $3i = 0$ and $3j = 0$. Hence, $3 \cdot (k+1) = 3i + 3j = 0 + 0 = 0$. Hence the property holds.

4. (10 points) Prove by induction that $3^n < n!$ for all integers n greater than 6. State whether you are using regular or strong induction.

5. (15 points) Prove by induction that every integer $n > 17$ can be written in the form $n = 4a + 7b$ where $a, b \geq 0$. State whether you are using regular or strong induction.

6. (10 points, 5 points each) Give a recursive definition of the following:

(a) the set of positive integers congruent to 4 modulo 5

(b) the sequence defined by $a_n = n(n+1)$ for $n \geq 1$

7. (15 points) Consider the subset S of the set of ordered pairs of integers defined recursively as follows:

   - Base case: $(0, 0) \in S$
   - Recursive case: If $(a, b) \in S$, then $(a+2, b+3) \in S$, and $(a+3, b+2) \in S$

   Based on this definition:

   (a) (5 points) List the first five elements in S

   (b) (10 points) Use structural induction to show that $5|(a+b)$ for all $(a, b) \in S$

8. (15 points) A bitstring is a string consisting of only 0's and 1's. Consider the following recursive definition of the function "count", which counts the number of 1's in the bitstring:

   - Base case: $\text{count}(\epsilon) = 0$
   - Recursive case 1: $\text{count}(1 \cdot s) = 1 + \text{count}(s)$
   - Recursive case 2: $\text{count}(0 \cdot s) = \text{count}(s)$

   Use structural induction to prove that $\text{count}(st) = \text{count}(s) + \text{count}(t)$.