Type your answers to the following questions and submit a PDF file to Blackboard. One page per problem.

Problem 1. [5pts] Consider the sequence $\{A_n\}_{n=0}^{\infty}$ where each element of the sequence is a set A_n, defined by $A_0 = \varnothing$ and for $n \geq 1$, $A_n = \{|A_i| : 0 \leq i < n\}$. Prove that $|A_n| = n$ for all $n \geq 0$.

Problem 2. [5pts] Define a sequence $\{c_n\}_{n=0}^{\infty}$ by $c_0 = 1$ and for all $n \geq 1$, let $c_n = \sum_{i=0}^{n-1} \frac{c_i}{2^{n-i}}$. Prove that for all $n \geq 1$, $c_n = \frac{1}{2}$. [Note: There are two ways to prove this statement. One is by strong induction. The other is to use weak induction after proving an equivalent recurrence relation. Either will be accepted.]

Problem 3. [10pts] The *merge sort algorithm* sorts a list of n numbers $x_1, \dots, x_n$. The algorithm works by first testing if $n = 1$, and if so does nothing. Otherwise, the algorithm recursively calls itself on the first $\lfloor n/2 \rfloor$ entries, and then calls itself on the last $\lceil n/2 \rceil$ entries, then "shuffles" them together by iterating through the two parts, selecting the minimum elements from each part until creating a sorted list of n entries. If t_n is the time it takes to run the merge sort algorithm on a list of n numbers, then $t_1 = 1$ (only need to test one operation), and (roughly)

$$t_n = 2t_{\lceil n/2 \rceil} + n$$

Using this recurrence relation and strong induction, prove that $t_n \leq 2n \log_2(n+1)$ for all $n \geq 1$.

Problem 4. [10pts] Let $S = (\mathbb{Q}, T, 0)$ be a state machine where the states are rational numbers ($\mathbb{Q}$) and the transitions are of the form $x \to x + 1$, $x \to 3x$, and $x \to \frac{x}{2}$, and the initial state is 0. Prove that no matter what transitions are used, the state $\frac{1}{3}$ will never be reached. [For 2 extra points: Prove that for any $\varepsilon > 0$, there is a sequence of transitions such that a state x is reachable where $|x - \frac{1}{3}| < \varepsilon$.]

Problem 5. [15pts] Let $\Sigma = \{0, 1\}$ (here we say Σ is the name of a set, not the summation notation). For $k \geq 0$, the set Σ^k is the set of k-tuples where every entry comes from Σ. The set Σ^* is equal to $\cup_{k=0}^{\infty} \Sigma^k$, the set of all finite binary strings (note that for every finite binary string $\mathbf{x}$ of length k, $\mathbf{x} \in \Sigma^k \subset \Sigma^*$). We will denote a string $\mathbf{x} = (x_1, x_2, \dots, x_k)$ as $x_1 x_2 \dots x_k$. Define a state machine $(\Sigma^*, T, 0)$ where the initial state is the string 0, and the transitions are of the form

$$x_1 x_2 \dots x_{k-1} 1 \to x_1 x_2 \dots x_{k-1} 10 \tag{1}$$
$$x_1 x_2 \dots x_{k-1} 0 \to x_1 x_2 \dots x_{k-1} 00 \tag{2}$$
$$x_1 x_2 \dots x_{k-1} 0 \to x_1 x_2 \dots x_{k-1} 01 \tag{3}$$
$$x_1 x_2 \dots x_{k-1} x_k \to x_k x_{k-1} \dots x_2 x_1 x_1 x_2 \dots x_{k-1} x_k. \tag{4}$$

Note that there are four types of transitions: (1) Take a string ending in 1 and add a 0, (2) take a string ending in 0 and add a 0, (3) take a string ending in 0 and add a 1, (4) take a string and add its reversal to the beginning (turning it into a palindrome).

a. Prove that the string 10011001 is reachable from the initial string 0.

b. Prove that the string 0110110 is not reachable from the initial string 0.

c. Prove that a reachable string can never contain three consecutive 1's.