**CS 420/520 Automata Theory**
**Fall 2023**

# Assignment 6

*due Thursday, November 30, 2023*

1. Give implementation-level descriptions of TMs that decide the following languages. (Describe in English or pesudo-code instead of a diagram.)
   (a) $\{ w \mid w$ contains twice as many 0s as 1s $\}$
   (b) $\{ w \mid w$ does not contain twice as many 0s as 1s $\}$

2. Carefully describe (give state diagram) a TM which will add one to the binary representation of a number. The number will have a \$ on the left end.
   - the number is written in reverse order: the number $13 = (1101)_2$ will be on the tape as \$1011.
   - If the input is the empty string, then the output should be \$.
   - if the input is \$, the output should be \$0
   - if the input is (for example) \$0101, the output should be \$1101, and \$111 should result in \$0001
   - trailing zeroes are acceptable (\$010 becomes \$110)
   - after correctly transforming the input, halt by entering the accepting state

3. What can a Turing machine with stay-put instead of left compute?

4. Let $A$ be a Turing-recognizable language consisting of descriptions of Turing machines $\{\langle M_1\rangle, \langle M_2\rangle, \ldots\}$, where every $M_i$ is a decider. Prove that some decidable language $D$ is not decided by any decider $M_i$ whose description appears in $A$. (Hint: you may find it helpful to consider an enumerator for $A$.)

5. **(grads)** *exercise 4.17 (2nd ed) or 4.18 (3rd ed):* Let $C$ be a language. Prove that $C$ is Turing-recognizable if and only if a decidable language $D$ exists such that

$$C = \{ x \mid \exists y \; (\langle x, y\rangle \in D) \}.$$

   *note:* In the text this is a starred (difficult) problem. It should not be, and is important in understanding the Turing-recognizable ($\triangleq$ recursively enumerable) languages. It has also an important analogy in the characterization of $NP$.
   *hint (for $\Rightarrow$):* Think of $y$ as the number of steps for which to simulate the TM for $C$.