Theory of Computation, CSCI 438, Spring 2016
Multitape Turing Machines and/or Nondeterministic TM, April 6

1. Use a multitape machine to recognize the language:
   \[ L = \{ w \mid w \in \{a, b\}^* \text{ and } n_a(w) = n_b(w) \} \]
   Begin with a high level plan. Next give a detailed plan and then the TM_{multitape}.

   **High level plan**
   Use a 3-tape machine – the input tape, a tape for the a’s and a tape for the b’s. Run through the input tape copying the a’s onto the a tape and the b’s onto the b tape. Travel back through the a’s and the b tape, making sure that they have the same number of elements.

   **Detail plan**
   Mark the front of the a and b tape with $.

   Travel right across the input tape, copying the a’s to the a tape and the b’s to the b tape. Stop when encounter a blank on the input tape.

   Travel left across the a and b tape until both reach a $, and accept. If one reaches a $ first, reject.

   **Machine**

   ![Machine Diagram]
Pages 178-180 describes a process of emulating a nondeterministic TM with a regular, deterministic TM. Look this over and answer the following questions.

2. Describe the process that happens each time a new simulation is started.
   a. Clear the simulation tape
   b. Copy the non-blank contents of the input tape onto the simulation tape
   c. Return the read/write head of all three tapes back to the beginning

3. As the simulation proceeds, describe what happens when an accept state is entered. Describe what happens when a reject state is entered. Describe what happens when arrive at a state with no transition to take.
   i. If the accept state is entered on the simulated non-deterministic Turing Machine, accept the string.
   ii. If the reject state is entered on the simulated non-deterministic Turing Machine, stop that simulation and do the next simulation. (It is likely that this branch will be taken many times so the reject state will be entered many times.)
   iii. If no transition, do the same as ii.

4. Describe the simulation process and how the progress tape is used.

   Loop:
   a. Prepare the tapes as described in the first question.
   b. Simulate the execution, using the number at the read/write head of the progress tape to tell which branch to take at each non-deterministic choice and moving the read/write head one position to the right.
   c. When a blank is encountered on the progress tape, update the progress tape

Updating the progress tape (a blank has been encountered)

Loop:
Move the read/write head of the progress tape one position to the left and attempt to increment the integer. If successful (all of the nondeterministic choices for this character and state have not been taken), the tape is updated. If the integer cannot be updated, set this value to 1 and go to the top of the loop.

If the start of progress tape is encountered, all branches at this level have been explored and future searches need to be one level deeper. To indicate this, travel right across the 1’s until reach blank. Replace the blank with a 1, indicating a new level.

The process of updating the progress tape requires a complex finite automaton.