**Week 10 Tutorial Session**

(1) In this problem, you will design Turing machine for the following language:

$$L = \{\mathtt{a}^n\#\mathtt{a}^n\#\mathtt{a}^n : n \geq 0\},\ \Sigma = \{\mathtt{a}, \#\}.$$

Give a high-level description of your Turing machine. If you have time left after finishing the next problem, give also a state diagram.

(2) A *queue automaton* is like a push-down automaton without the input tape and the stack is replaced by a queue. A queue is a tape allowing symbols to be read only at the left end and written only at the right end. At each time step, the queue automaton may perform either a read or write operation. Each read operation (called a *pop*) reads and removes a symbol from the left and of the tape and each write operation (called a *push*) writes a symbol at the right end. For example, if the state of the tape is `abcaaab`, the operation **pop** `a` yields `bcaaab`. Now **push** `c` yields `bcaaabc`. The internal state of the queue automaton may change after each *pop* or *push* operation (and this transition may depend on the symbol pushed or popped). Initially, the queue contains the input followed by the special end-of-input symbol `$`. The automaton accepts (rejects) by going into a special state $q_{\text{accept}}$ ($q_{\text{reject}}$). The transitions in a queue automaton are deterministic. You will argue that a queue automaton is equivalent to a Turing machine: Every queue automaton can be simulated on a Turing machine, and vice versa.

   (a) Write a formal definition of a queue automaton. A formal definition of an automaton will look like page 17 of Lecture 14 or page 9/slide 7 of Lecture 10.

   (b) Show how to simulate a queue automaton on a Turing machine. For this, you need to specify

   - how the tape of the Turing machine will be used to represent the queue automaton;
   - how the Turing machine tape should be set up initially;
   - what the Turing machine should do when the automaton performs a **push** or a **pop** (you may specify in 1-2 sentences the general idea, omitting the tedious details);
   - what the Turing machine should do when the queue automaton accepts/rejects.

   (c) Show how to simulate a Turing machine on a queue automaton. For concreteness, you may assume the Turing machine has tape alphabet $\Gamma = \{\mathtt{a}, \mathtt{b}, \square\}$. Again you should specify simulation details similar to those in part (b).