Cpt S 317 Homework #10

Please print your name!

1. Build a Turing machine accepting $(b + c)^+ \# a^+$. (Please comment your code. Any uncommented solutions will not be graded.)

2. Build a Turing machine accepting $\{x \# x^r : x \in \{a, b\}^+\}$. (Please comment your code. Any uncommented solutions will not be graded.)

3. Turing machines are definitely stronger than PDAs (why?). But it is interesting to investigate some restricted forms of TMs and see if the restrictions we apply could maintain the computing power of TMs. Here is an example. Consider a restricted TM M, called a *one-turn* Turing machine, such that during any executions on any input, M makes at most one turn on the tape (i.e., the tape head either always moves to the right, or moves to the left later but never moves to the right again. However, the tape head could stay at any moment.) Show (describe) that languages accepted by one-turn Turing machines are context-free.

4. Many of you asked why we couldn't have two stacks for PDAs? Now, here comes a problem for you. It is natural to add an extra stack to a PDA. Say M is a *two-stack PDA* if M is exactly the same as a PDA but with two stacks. Each instruction (transition) in M is in the form of

$$(p, \gamma_1, \gamma_2) \in \delta(q, a, b_1, b_2)$$

which means that if M is at state q and reading input symbol a with the top of the two stacks being b_1 and b_2 respectively, then this transition brings M to state p, and replaces the tops b_1 and b_2 of the two stacks to γ_1 and γ_2 respectively. Show (describe) that any Turing machine can be simulated by a two-stack PDA.