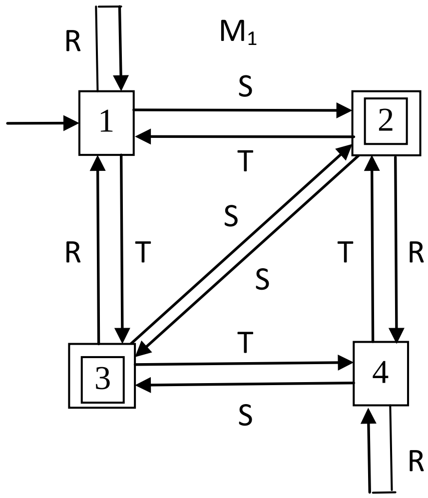
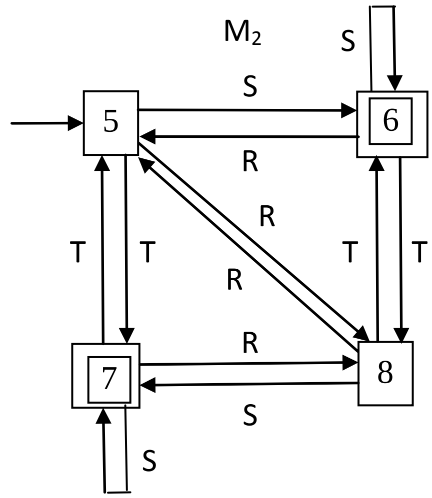
**CS 475/575 Exam 1** **Spring 2011** **Solution**

1. For each of these finite-state machines $M_1$ and $M_2$, write an accepting computation sequence using input string RSRSTT.




**$M_1$: $1 \rightarrow^{R} 1 \rightarrow^{S} 2 \rightarrow^{R} 4 \rightarrow^{S} 3 \rightarrow^{T} 4 \rightarrow^{T} 2$.**

**$M_2$: $5 \rightarrow^{R} 8 \rightarrow^{S} 7 \rightarrow^{R} 8 \rightarrow^{S} 7 \rightarrow^{T} 5 \rightarrow^{T} 7$.**

2. Prove by mathematical induction that the two finite-state machines given in problem 1 are equivalent; that is, for every input string w of every length $n \geq 0$, prove $w \in L(M_1)$ iff $w \in L(M_2)$.

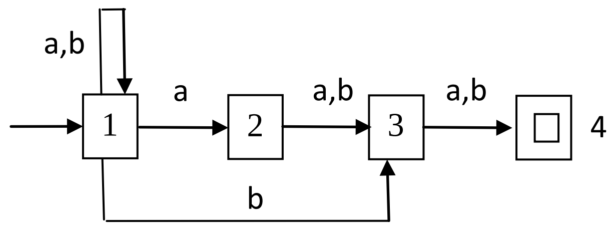
**<u>Basis</u>: When the length n=0, then w = $\varepsilon$, which is rejected by both $M_1$ and $M_2$.**
**<u>Inductive hypothesis</u>: Choose $n \geq 1$, and suppose for every string w of length n – 1, that $w \in L(M_1)$ iff $w \in L(M_2)$.**
**<u>Inductive step</u>: Let w be any string of length $n \geq 1$. We can write w = yc, where y is a string of length n – 1, and c is an input character. By the inductive hypothesis, $y \in L(M_1)$ iff $y \in L(M_2)$.**
**So now there are three cases for input character c:**

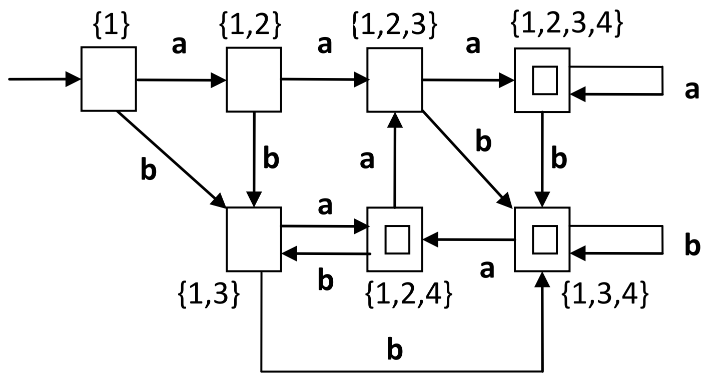
- **If c = R, then w = yR, which is always rejected by both $M_1$ and $M_2$.**
- **If c = S, then w = yS, which is always accepted by both $M_1$ and $M_2$.**
- **If c = T, then w = yT, and this case has two subcases:**
  - **If y is accepted by both $M_1$ and $M_2$, then w is rejected by both $M_1$ and $M_2$.**
  - **If y is rejected by both $M_1$ and $M_2$, then w is accepted by both $M_1$ and $M_2$.**

**In all cases, $w \in L(M_1)$ iff $w \in L(M_2)$, so $L(M_1) = L(M_2)$.**

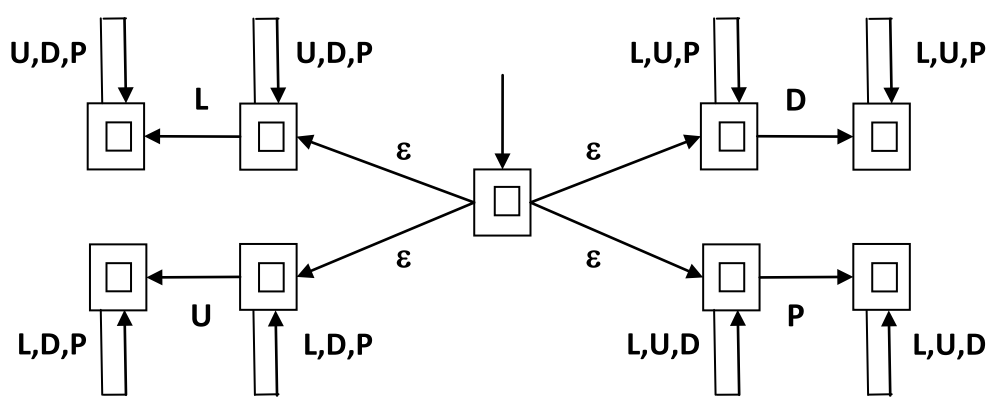
**Intuition (not required to write the proof):**
**R = reset to non-final state, S = set to final state, T = toggle between non-final and final.**

3. Draw a deterministic finite-state machine that is equivalent to the following non-deterministic finite-state machine:

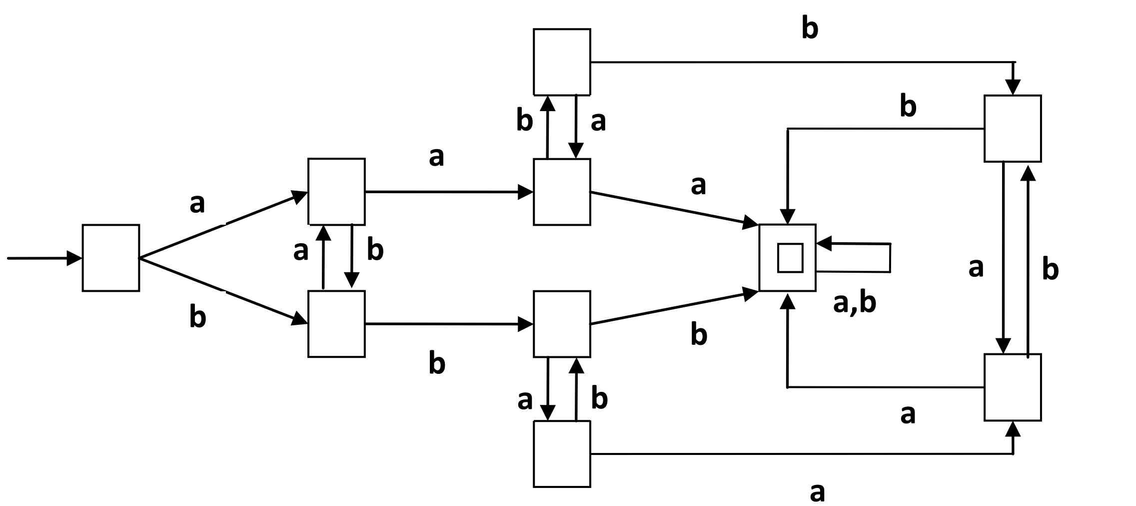



4. Let input alphabet {L, U, D, P} represent these four categories: lowercase, uppercase, digit, punctuation. Define a *strong* password as any string that contains at least two symbols from each category. Also define a *weak* password as any string that is not a strong password. Draw a *non*-deterministic finite-state machine that accepts *weak* passwords.



5. Draw a deterministic finite-state machine that accepts the set of strings over alphabet {a,b} that contain at least one of the following: the substring aaa, or the substring bbb, or two aa substrings, or two bb substrings.




6. Draw a deterministic finite-state machine that accepts the set of valid identifiers in a programming language in which each identifier can be any non-empty string of lowercase letters, other than these specified reserved words: {if, iff, of, off, on, one}.