Harvard University
Computer Science 121

Section 1 Handout
Week of 9.21.09

1 Administrative matters

- Office Hours
- Problem set 1 due Friday – basement of Maxwell-Dworkin. Please staple each part separately.

2 DFA's

- Formal vs. informal descriptions: informally, a DFA is described by a state diagram, as seen in Sipser. Formally, a DFA is a 5-tuple $(Q, \Sigma, \delta, q_0, F)$ where Q is a finite set of states, Σ is an alphabet, $\delta : Q \times \Sigma \to Q$, $q_0 \in Q$ is the start state, and $F \subseteq Q$ is the set of final (accepting) states. We say that $(Q, \Sigma, \delta, q_0, F)$ *accepts* a string $w = \sigma_1 \cdots \sigma_n$ if and only if, for some sequence of states $q_0, q_1, \cdots, q_n \subseteq Q$, we have $q_n \in F$ and for each $i \in [0, n)$, $\delta(q_i, \sigma_{i+1}) = q_{i+1}$.
- Write a DFA that recognizes the language of all strings ending in b.
- Write a DFA that recognizes the language (over $\Sigma = \{0, 1\}$) of all binary numerals that give a remainder of 3 when divided by 4.

3 NFA's

- Motivation: we want a machine that can 'guess'.
- Precise difference from DFA's: $\delta : Q \times (\Sigma \cup \{\varepsilon\}) \to \mathcal{P}(Q)$ – i.e., there is a *set* of permissible states, instead of exactly one, and we can have *epsilon-transitions* (transitions on the empty string).
- Write an NFA that recognizes the language of all strings ending in ab.

- Write an NFA that recognizes the language of all strings with the same first and last symbol.

- Show that the class of regular languages is closed under intersection.

- Define INTERLEAVE$(L_1, L_2) = \{w = a_1 b_1 a_2 b_2 ... a_n b_n \mid a_i \in L_1 \text{ and } b_i \in L_2 \;\; \forall 1 \geq i \geq n\}$
  Show that the regular languages are closed under the operation INTERLEAVE

4 Proof by Counterexample

In particular, let us talk about counterexamples. See if you can prove or disprove *All even square numbers are divisible by four.* Now see if you can prove or disprove *All prime numbers are odd.*

Notice the difference? To prove a statement that quantifies over "all elements" with a certain property, you need to give an argument that works for every element, but to refute such a statement, one simple example suffices.

Now that you've thought about this, let us do some examples. Prove or disprove the following statements.

- If L_1 and L_2 are both infinite languages, then $L = \{w_1 w_2 \mid w_1 \in L_1 \text{ and } w_2 \in L_2\}$ contains a string that is not in L_1 or L_2.

- If L is a finite language, $L \subsetneq L^*$.

- If L is an infinite language $L \subsetneq L^*$.