CSE 531: Computability and Complexity
Assignment #**0**
September 29, 2005

Autumn 2005
Instructor: Paul Beame

Does NOT have to be turned in.

Course webpage: `http://www.cs.washington.edu/531`
Important: Subscribe to CSE 531 email group ASAP by visiting
`http://majordomo.cs.washington.edu/mailman/listinfo/cse531`.

Note: The goal of this problem set is to aid you in reviewing material on finite automata, regular languages, and context-free grammars. Though we will not cover this material in lecture, background on these topics is useful for some material we will cover in the class. If you don't remember how these arguments work or didn't see them in your undergrad course, this problem set will also serve as a warm-up for the skills you will need in this class.

---

1. For any string $w = w_1 w_2 \ldots w_n$ for each $w_i \in \Sigma$, the *reverse* of w, written $w^{\mathcal{R}}$, is the string $w_n \ldots w_2 w_1$ and $A^{\mathcal{R}} = \{w^{\mathcal{R}} \mid w \in A\}$. Show that if A is regular, then so is $A^{\mathcal{R}}$.)

2. (Part of the Myhill-Nerode theorem) Let Σ^* be an alphabet and $L \subseteq \Sigma^*$. Say that two strings $x, y \in \Sigma^*$ are L-equivalent if and only if for every string $z \in \Sigma^*$, either both xz and yz are in L or neither is in L.
   (a) Show that for any DFA M, if two strings x and y lead to the same state of M then x and y are $L(M)$-equivalent.
   (b) Use part (a) to show that if there exist $x_1, \ldots, x_k \in \Sigma^*$ such that no two distinct x_i and x_j are L-equivalent then any DFA M with $L(M) = L$ requires at least k states.

3. (Exponential blow-up is necessary when converting NFAs to DFAs.) Let $E_n = \{x \in \{0,1\}^* : x$ has a 1 in the n-th position from the right $\}$. Show that the any NFA for E_n has $O(n)$ states but that any DFA requires at least c^n states for some $c > 1$. (For the latter part use the result of the previous problem.)

4. Give (terminating) algorithms for answering the following questions about finite automata. (For a DFA M, $L(M)$ is the language of strings accepted by M.)
   (a) Given a DFA M, is $L(M) = \Sigma^*$ (i.e. does M accept *every* string)?
   (b) Given two DFAs M_1 and M_2, is $L(M_1) \subseteq L(M_2)$?
   (c) Given two DFAs M_1 and M_2, is $L(M_1) = L(M_2)$?

5. Prove that the intersection of a context-free language with a regular language is context-free.

6. Give an example to show that context-free languages are **not** closed under intersection. Deduce that CFLs are not closed under complementation.

7. Given a context-free grammar G, give an algorithm to convert G to Chomsky normal form.

8. Given a context-free grammar G, give an algorithm to decide whether or not $L(G) = \emptyset$.

9. Give a context-free grammar for the language $A = \{a^i b^j c^k \mid i = j \text{ or } j = k \text{ and } i, j, k \geq 0\}$. Show that your grammar is ambiguous.