CSE 431
Spring Quarter 2001
Assignment 4
Due Friday, April 27

All solutions should be neatly written or type set. All major steps in proofs and algorithms must be justified.

1. (10 points) Consider the following problem:

Input:	A one tape Turing machine M and a symbol σ in its tape alphabet.
Property:	On some input M writes the symbol σ on its tape.

   Use a reduction to show that the problem is undecidable.

2. (10 points) Consider the Nonempty Intersection Problem for Context-Free Grammars:

Input:	Two context-free grammars G_1 and G_2
Property:	$L(G_1) \cap L(G_2)$ is nonempty.

   We can use a reduction using accepting histories to show that this problem is undecidable. In this case an accepting history $C_1, C_2, \ldots C_m$ of M on input w is represented by the string

   $$C_1\#C_2^R\#C_3\#C_4^R\# \cdots \#C_m$$

   assuming that m is an odd number. We can alway assume that if M accepts w then it does so in an odd number of configurations. One can design a context-free grammar G_1 that generates the language

   $$L(G_1) = \{C_1\#C_2^R\#C_3\#C_4^R\# \cdots \#C_m : C_i \text{ yields } C_{i+1} \text{ for } i \text{ even and } C_1 = q_0 w \sqcup^k \text{ for some } k\}.$$

   The easiest way to think about this is to see how a PDA can accept $L(G_1)$. The PDA would first check that the string up to the first $\#$ is in $q_0 w \sqcup^*$. This is a regular language so the PDA does not even need its stack. At this point the PDA would push the string up to the second $\#$ on to the stack. It would then compare the string up to the third $\#$ to the string stored on the stack to see if the string stored on the stack yields the string on the input. This checking can be done because there is only a finite area on the two strings that are different. Where they are different (near the state symbol) the PDA can check that the rules of the Turing machine were followed. This process is continued for every pair of consecutive strings where the first appears just after an odd number of $\#$'s. The PDA can then be converted to the context-free grammar G_1 using the equivalence of PDA's and context-free grammars.

   Define carefully the language $L(G_2)$ so that M accepts w if and only if $L(G_1) \cap L(G_2)$ is nonempty. Briefly describe how a PDA can accept the language $L(G_2)$.