CSE 431: Introduction to Theory of Computation Spring 2005

PROBLEM SET 5

Due Friday, May 13, 2005, in class

Reading assignment: Sipser's book, Sections 7.1-7.3.

Instructions: Same as for Problem set 1.

Each question is worth 10 points. Please be as clear and concise as possible in your arguments and answers. The optional problem is for extra credit.

---

1. (a) Show that the Post Correspondence Problem is decidable over a unary alphabet, i.e., over the alphabet $\Sigma = \{1\}$.

   (b) Show that the Post Correspondence Problem is undecidable over a binary alphabet, i.e., over the alphabet $\Sigma = \{0, 1\}$.

2. (a) Problem 5.19, Sipser's book (Ambiguity of CFGs is undecidable)

   (b) Use the approach used in part (a) above to give a proof different from the one given in the previous problem set of the undecidability of $\mathit{DISJOINT}_{\text{CFG}}$ defined as:

   $$\mathit{DISJOINT}_{\text{CFG}} = \{\langle G_1, G_2\rangle \,|\, G_1, G_2 \text{ are context-free grammars and } L(G_1) \cap L(G_2) = \emptyset\} \, .$$

3. Let

   $$\mathit{MODEXP} = \{\langle a, b, c, p\rangle \mid a, b, c, \text{ and } p \text{ are binary integers such that } a^b \equiv c \pmod{p}\} \, .$$

   Show that $\mathit{MODEXP} \in \mathrm{P}$.

4. Show that P is closed under the star operation. (Hint: Use dynamic programming.)

5. * **(Optional Problem, Tricky!)** Let $f : \mathbb{N} \to \mathbb{N}$ be any function where $f(n) = o(n \log n)$. Show that $\mathrm{TIME}(f(n))$ contains only regular languages.
   (Suggestion: The key concept that aids showing the above is that of a *crossing sequence*. When a TM is run on an input, the crossing sequence at a given cell is the sequence of states that the machine enters at that cell as the computation progresses. Now, expand on the following two high-level ideas concerning crossing sequences. First, show that for a particular TM, if all crossing sequences on all inputs are of a fixed length ℓ or less, once can simulate the TM by an NFA. Second, show that a TM with unbounded crossing sequence length cannot run in $o(n \log n)$ time. For this, use a counting/pigeonholing argument to deduce repetition of crossing sequences at multiple places and then get a contradiction by "splicing" a minimal length input to a smaller string on which the TM has identical behavior.)