15-453 Formal Languages, Automata, and Computation

Homework 7, due March 17

Problem 1 (30 points)

Using the inequality symbol for the mapping (or many-one) reducibility relation ($\leq_m$) makes some sense, because this relation behaves in many ways like a partial order relation. Prove:

1. $\leq_m$ is reflexive. I.e. $A \leq_m A$.
2. [problem 5.6 in the book] $\leq_m$ is transitive. I.e. if $A \leq_m B$ and $B \leq_m C$ then $A \leq_m C$.
3. In general $\leq_m$ is not symmetric: $A \leq_m B$ does not imply $B \leq_m A$.

Give **clear formal proofs** to accompany your arguments.

Problem 2 (40 points)

Let M be a TM over alphabet Σ with states Q.

1. Show that the following language is context-free:

$$\{\#xqy\#(wq'z)^R \mid xqt \Rightarrow wq'z, \{x, y, w, z\} \subset \Sigma^*, \{q, q'\} \subseteq Q\}$$

(i.e. xqy and $wq'z$ can be consecutive snapshots of M's configuration during a computation of M).

2. Show that valid accepting computation histories of M, described by strings of the form $\#c_1\#c_2^R\#c_3\ldots\#c_n$ can be expressed as the intersection of two context-free languages. Each c_i is a string of the form xqy, as in 1. **Hints:** See page 182 in the book. Use 1 and the fact that CFLs are closed under the regular operations.

3. Prove that the language

$$\{\langle G_1, G_2\rangle \mid G_1, G_2 \text{ are CFGs and } \mathcal{L}(G_1) \cap \mathcal{L}(G_2) \neq \phi\}$$

is undecidable. **Hints:** Use 2 and a reduction from A_{TM}.

Be as formal and clear in your solutions. Very informal arguments will not receive full-credit.

Problem 3 (30 points)

[Problem 5.9 from the book]

Prove that any recognizable language reduces (using $\leq_m$) to A_{TM}.

Note: this statement is sometimes read as "A_{TM} is *complete* for the set of Turing-recognizable languages with respect to $\leq_m$".

Extra credit: Problem 4 (30 points)

We define a class of automata called RO-TM. These behave like normal TMs, but their (unique) tape is read-only. Let M be such a machine, with $q = |Q|$ states, operating on an input word w.

1. Show that there exists a constant c_M, such that no computation history of size greater than $l(n) = c_M q(q + n)$ can describe an accepting computation of w with $n = |w|$.

2. Conclude that the halting problem for RO-TMs is decidable.

3. For Turing machines, we can similarly define

$$l(n) = \max_{\substack{w \in \mathcal{L}(M) \\ |w|=n}} \{\text{length of a computation history of } M \text{ on } w\}.$$

Why can't we apply the same argument as in 2?