Problem Set 7 – Due Tuesday, March 6, 2012

Warning: this is a long but important homework.

Problem 1. Classify each of the following languages as either (a) recursive—I see how to decide this language; (b) r.e.—I don’t see how to decide this language, but I can see a procedure to accept this language; (c) co-r.e.—I don’t see how to decide this language, but I can see a procedure to accept the complement of the language; or (d) neither: I don’t see how to accept this language nor its complement. No justification is needed for your answers.

Part A. \{⟨M⟩ : M is a TM that accepts some string of prime length\}.

Part B. \{⟨M⟩ : M is a TM and M has 100 states\}.

Part C. \{⟨M⟩ : M is a TM and L(M) = L(M)^r\}.

Part D. \{⟨M⟩ : M is a TM and L(M) is r.e. \}.

Part E. \{⟨M⟩ : M is a C-program that halts on ⟨M⟩\}.

Part F. \{⟨M⟩ : M is a TM and M will visit state q_{20} when run on some input x\}.

Part G. \{⟨M⟩ : M is a TM and M uses at most 20 tape cells when run on blank tape\}.

Part H. \{(G) : G is a CFG and G accepts an odd-length string\}.

Part I. \{(G_1,G_2) : G_1 and G_2 are CFGs and L(G_1) = L(G_2)\}.

Problem 2. Prove whether each of the following languages is recursive, r.e. but not recursive, co-r.e. but not recursive, or neither r.e. nor co-r.e.

Part A. \{⟨M,w⟩ : M is a TM that uses at most 20 tape squares when run on w\}.

Part B. \{⟨M,k⟩ : M is a TM that accepts at least one string of length k\}.

Part C. \{⟨M,k⟩ : M is a TM that diverges (loops) on at least one string of length k\}.

Part D. \{⟨M,k⟩ : M is a TM that accepts a string of length k and diverges on a string of length k\}.

Assume that the underlying alphabet has at least two characters.

Part E. \{⟨M⟩ : M is a TM that accepts some palindrome\}.

Problem 3 Say that a language \(L = \{x_1,x_2,\ldots\}\) is enumerable if there exists a two-tape TM \(M\) that outputs \(x_1x_2x_3\cdots\) on a designated output tape. The other tape is a designated work tape, and the output tape is write-only, with the head moving only from left-to-right. Say that \(L\) is enumerable in lexicographic order if \(L\) is enumerable, as above, and, additionally, \(x_1 < x_2 < x_3 < \cdots\), where “<” denotes the usual lexicographic ordering on strings.

Part A. Prove that \(L\) is r.e. iff \(L\) is enumerable. (This explains the name “recursively enumerable.”)

Part B. Prove that \(L\) is recursive iff it is enumerable in lexicographic order.

Problem 4∗ Challenging. An unrestricted grammar \(G = (V,\Sigma,R,S)\) is like a CFG except that rules have lefthand sides from \((\Sigma \cup V)^*V(\Sigma \cup V)^*\). Whenever you have a rule \(\alpha \rightarrow \beta\), you can replace \(\alpha\), wherever it occurs in a sentential form \(\sigma\) within a derivation, with \(\beta\). The language of an unrestricted grammar \(G\) is, as usual, the set of terminal strings derivable from the start symbol: \(L(G) = \{x \in \Sigma^* : S \Rightarrow x\}\). Show that the languages of unrestricted grammars are exactly the r.e. languages.