Math 27800 / CS 27800, Winter 2024: Assignment 3
Due Friday, February 2nd

In the following problems, when describing computable procedures, it is enough to give informal (but precise) descriptions of your algorithms. You definitely should not try to define Turing machines implementing these algorithms... (In particular, to describe a computably enumerable set, it is enough to give an informal algorithm for listing its elements.)

In the hints below, $\Phi_0, \Phi_1, \ldots$ is an effective listing of all partial computable $0,1$-valued functions. (We defined such a listing for all partial computable functions, but we can do the same for $0,1$-valued functions just by interepreting any function value greater than 0 as 1.) By *effective listing* we mean that the function $\langle e, n\rangle \mapsto \Phi_e(n)$ is itself partial computable.

Problem 3 is a "take-home exam" problem, to be worked out individually.

1. Let $\sigma_0, \sigma_1, \ldots$ list the finite binary strings in length-lexicographic order, and say that a set of binary strings S is computable if $\{i : \sigma_i \in S\}$ is computable.

A *binary tree* is a set T of finite binary strings such that if $\sigma \in T$ and $\tau \prec \sigma$ (i.e., τ is an initial segment of σ) then $\tau \in T$. An *infinite path* on a tree T is an infinite binary sequence α such that every initial segment of α is on T.

a. Show that every infinite binary tree has an infinite path.

b. Show that there is a computable infinite binary tree with no computable infinite path. [Hint: We can think of each total Φ_e as an infinite binary sequence. Build T so that for each such Φ_e, there is some initial segment of Φ_e that is not in T.]

c. Let T be a computable infinite binary tree. Show that if we could compute the Halting Problem, then we could compute an infinite path on T. [Here it might be helpful to use the following fact, which you may assume: Let g be a partial computable binary function. There is a partial computable function s such that for all x, y, we have $\Phi_{s(x)}(y) = g(x, y)$.]

d. Show that there is a computable tree T such that, if we could compute an infinite path on T, then we could compute a completion of ZFC (i.e., a complete, consistent theory extending ZFC). [Note: ZFC is used here only as an example; this would work for any theory with a computable set of axioms. You may assume the fact that the set of axioms of ZFC is computable.]

2. Show that there are computably enumerable sets A and B that are *computably inseparable*, meaning that $A \cap B = \emptyset$ but there is no computable set C with $A \subseteq C$ and $B \cap C = \emptyset$. [Hint: Build A and B so that for each e, if Φ_e is total then there is an n such that either $\Phi_e(n) = 0$ and $n \in A$ or $\Phi_e(n) > 0$ and $n \in B$.]

3 (take-home exam problem). Let A and B be computably enumerable subsets of $\mathbb{N}$. For each of the following sets, must the set be c.e.? (In each case, prove or give a counterexample.): $A \cup B$, $A \cap B$, $A \setminus B$.