CS 517
Due: Wed 13 Apr

Homework #2

1. Consider a listing of the constant functions, e.g.

$$f_0(x) = 0, \quad f_1(x) = 1, \quad f_2(x) = 2, \quad \ldots.$$

   Diagonalize over this listing to produce a NON-constant function What is this diagonal function?

2. Argue that if

$$g_0(x), \quad g_1(x), \quad g_2(x), \quad \ldots$$

   is a listing of the constant functions then the diagonal function for this listing is a non-constant function.
   BIG Question: If $d(n)$ is the diagonal function, can you say anything aabout its growth rate? E.g. is $d(n) = \Omega(n)$?

3. Show that if

$$p_0(x), \quad p_1(x), \quad p_2(x), \quad \ldots$$

   is a *computable* listing of polynomials then the diagonal function is a COMPUTABLE function which is NOT a polynomial.

4. Show that if

$$f_0(x), \quad f_1(x), \quad f_2(x), \quad \ldots$$

   is a *computable* listing of any class of computable functions then the diagonal function is a COMPUTABLE function which is NOT in this class.

5. Do Ex 1.2 from the Notes.

6. Do Ex 1.4 from the Notes.

7. Do Ex 1.6 from the Notes.

8. In-Equality of Primitive Recursive Functions
IN-EQ-PRIM :
INPUT: Two primitive recursive functions $f_1(x)$ and $f_2(x)$.
(Actually we assume that you are given the primitive recursive programs for computing f_1 and f_2.)
QUESTION: Is $f_1 \neq f_2$?
(I.E., does there exist a v so that $f_1(v) \neq f_2(v)$?)

(a) Show that the set of **YES** instances of this problem is in **RE**.

(b) Show that if you had a recognizer for this set, then you could build a recognizer for the *HALT* set. (**HINT:** You may assume that running a program for x steps is a primitive recursive function of x.)

(c) Conclude that there is **NO** recognizer for the set of **YES** instances of **IN-EQ-PRIM**, i.e. this problem is algorithmically unsolvable.