Due Date: Feb. 11, 2003

Problem 1: Let L be a language accepted by a k-tape nondeterministic TM in time $f(n)$. Show that there exists a 2-tape nondeterministic TM that also accepts L in time $k \cdot f(n)$.

Note that the running time can be improved to $f(n)$ using the linear speed-up theorem since k is a constant (assuming $f(n) > cn$ for some constant $c > 2$.

(**Hint:** *Guess the computation and verify it.*)

Problem 2:

(i) Let L be r.e. language. Show that there exists a deterministic TM that enumerates the strings of L.

(ii) Show that a language L is recursive if and only if there is a Turing machine M that enumerates the strings of L in a nondecreasing order of their lengths.

Problem 3:

- Is $L = \{\langle M \rangle \mid L(M) \text{ is finite.}\}$ r.e.?
- Let $L = \{\langle M \rangle \mid M \text{ halts on all inputs.}\}$. Is L r.e.? What about $\overline{L}$.

Problem 4: Let Σ be some alphabet, and let $A \subseteq \Sigma^*$. A is called *productive* if there is a recursive function g such that for any Turing machine M if $L(M) \subseteq A$, then $g(\langle M \rangle) \in A \setminus L(M)$. (For example, $\overline{H}$ is productive, with $g(x) = x$.) Show that if A is productive, then A contains an infinite r.e. subset.

Problem 5:

(i) Show that $f(n) = 2^{g(n)} \cdot 3^{h(n)}$ is primitive recursive, assuming that $g(n)$ and $h(n)$ are primitive recursive.

(ii) let $\varphi(n)$ be the *Fibonacci function* defined as

$$\begin{aligned} \varphi(0) &= 0, \\ \varphi(1) &= 1, \\ \varphi(n+2) &= \varphi(n) + \varphi(n+1) \quad \forall n \in \mathbb{N} \end{aligned}$$

Show that $\varphi(n)$ is primitive recursive.

(**Hint:** *Use (i) to handle the double recursion.*)