Complexity Theory
Lent 2003
Suggested Exercises 4

1. Given a graph $G = (V, E)$, a set $U \subseteq V$ of vertices is called a *vertex cover* of G if, for each edge $(u, v) \in E$, either $u \in U$ or $v \in U$. That is, each edge has at least one end point in U. The decision problem V-COVER is defined as:

   > given a graph $G = (V, E)$, and an integer K, does G contain a vertex cover with K or *fewer* elements?

   (a) Show a reduction from IND to V-COVER.

   (b) Use (a) to argue that V-COVER is NP-complete.

2. The problem of four dimensional matching, 4DM, is defined analogously with 3DM:

   > Given four sets, W, X, Y and Z, each with n elements, and a set of quadruples $M \subseteq W \times X \times Y \times Z$, is there a subset $M' \subseteq M$, such that each element of W, X, Y and Z appears in exactly one triple in M'.

   Show that 4DM is NP-complete.

3. Define a *strong* nondeterministic Turing machine as one where each computation has three possible outcomes: accept, reject or maybe. If M is such a machine, we say that it accepts L, if for every $x \in L$, every computation path of M on x ends in either accept or maybe, with at least one accept *and* for $x \notin L$, every computation path of M on x ends in reject or maybe, with at least one reject.

   Show that if L is decided by a strong nondeterministic Turing machine running in polynomial time, then $L \in \mathsf{NP} \cap \text{co-}\mathsf{NP}$.

4. We use $x; 0^n$ to denote the string that is obtained by concatenating the string x with a separator ; followed by n occurrences of 0. If $[M]$ represents the string encoding of a *non-deterministic* Turing machine M, show that the following language is NP-complete:

$$\{[M]; x; 0^n \mid M \text{ accepts } x \text{ within } n \text{ steps}\}.$$

   Hint: rather than attempting a reduction from a particular NP-complete problem, it is easier to show this from first principles, i.e. construct a reduction for any NDTM M, and polynomial bound p.

   Similarly, if $[M]$ represents the encoding of a *deterministic* Turing machine M, then

$$\{[M]; x; 0^n \mid M \text{ accepts } x \text{ within } n \text{ steps}\}.$$

   is P-complete.

5. Define a *linear time reduction* to be a reduction which can be computed in time $O(n)$.

   (a) Show that there are no problems complete for P under linear time reductions (hint: use the Time Hierarchy Theorem).

   (b) Show that for any fixed k, there is a polynomial time decidable language L, such that every language in $\mathsf{TIME}(n^k)$ is reducible to L (hint: construct a language similar to the one in (4) above).