CSC363 - Summer 2007
Assignment 3

Problem 1 [15p]

For each of the following languages, state whether it belongs to P, NP or $coNP$. Make the strongest claim you can and justify it by describing an appropriate algorithm or verifier. Note: for this problem, note that we are explicitly encoding some values (t) in base 1. Everything else is encoded efficiently, i.e. base 2.

1. $\{\langle N, w, 1^t \rangle : N$ is a NTM and it accepts input w witin t steps $\}$.
2. $\{\langle M, w, 1^t \rangle : M$ is a regular TM and it accepts input w witin t steps $\}$.
3. $\{\langle G, k \rangle : G$ is a graph and it does not contain a clique of size $k\}$.

Problem 2 [10p] Consider the following problem: given n and t, can we write n as the product of t prime numbers? We formalize this problem as a language membership problem as follows. Define

$$\text{NUMPRIMEDIVS} = \{\langle n, t \rangle : n = p_1 \cdot \ldots \cdot p_t \text{ where for all } i, p_i \text{ is a prime number}\}$$

For example, $100 = 2 \cdot 2 \cdot 5 \cdot 5$, so $\langle 100, 4 \rangle \in$ NUMPRIMEDIVS, but $\langle 100, 3 \rangle$ and $\langle 100, 5 \rangle$ are not.

Show that NUMPRIMEDIVS is in NP.

Problem 3 [20p] Imagine that we allow the running time of a verifier to be a function of the length of the certificate. A verifier V works in polytime under this new definition if there exists a constant k such that the running time of V is at most $O((|x| + |c|)^k)$ (rather than $O(|x|^k)$ as before).

Show that a language L has a polytime verifier under the new definition iff L is recognizable.

Problem 4 [20p] A formula in CNF is *monotone* if it does not contain any negated variable. For example,

$$(x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee x_4) \wedge (x_2 \vee x_4)$$

Monotone formulas are always satisfiable (by setting all variables to true) but we can ask if it is possible to satisfy a monotone formula by setting fewer variables to true (e.g. the formula above can be satisfied by setting x_1 and x_4 to true, but not by setting only one variable to true).

Consider the following problem: given a monotone CNF f and some k, is there a satisfying assignment for f that sets at most k variables to true? We formalize this problem as a language membership problem as follows. Define MONOTONESAT $= \{\langle f, k \rangle : f$ is a monotone CNF formula that can be satisfied by setting at most k variables to true.

1. Show that VERTEXCOVER $\leq_p$ MONOTONESAT.
2. Show that SAT $\leq_p$ MONOTONESAT.