CMPT 308 - Computability and Complexity: Homework 4
(Due: Nov 29)

November 18, 2016

Reminder: the work you submit must be your own.

1. **Closure of NP**
   Show that the class NP is closed under the Kleene star operation (i.e., for any language L, if $L \in \mathsf{NP}$, then $L^* \in \mathsf{NP}$ as well).

2. **NP-completeness**

   (a) Define Root-CLIQUE = $\{\langle G \rangle \mid$ graph G has a clique of size at least $\sqrt{n}$, where n is the number of vertices in $G\}$. Prove that Root-CLIQUE is NP-complete, using a reduction from CLIQUE.

   (b) Define Twice-SAT = $\{\langle \phi \rangle \mid \phi$ is a cnf formula with at least two satisfying assignments$\}$. Prove that Twice-SAT is NP-complete, using a reduction from SAT.

   (c) Define PARTITION $= \{\langle a_1, \ldots, a_n \mid a_1, \ldots, a_n$ are positive integers in binary such that there is partition of $\{1, \ldots, n\}$ into two disjoint subsets S and T, where $S \cup T = \{1, \ldots, n\}$, so that $\sum_{i \in S} a_i = \sum_{j \in T} a_j\}$. Prove that PARTITION is NP-complete, using a reduction from SubsetSum.

3. **PSPACE-completeness**
   For a language A, an A-oracle Turing machine is a Turing machine M that may ask if $y \in A$, for any string y, and receive the correct answer in a single step. (That is, for such an A-oracle machine, checking membership in the language A is free.) We say that a language L is in P^A, if there is a deterministic polytime A-oracle TM that decides L. Similarly, we say that $L \in \mathsf{NP}^A$ if there is a nondeterministic polytime A-oracle TM deciding L.

   Prove that $\mathsf{NP}^{\mathrm{TQBF}} = \mathsf{P}^{\mathrm{TQBF}}$, where TQBF is the language of true quantified boolean formulas (which we showed in class to be PSPACE-complete).

4. **Randomized complexity**
   Suppose that $\mathsf{SAT} \in \mathsf{BPP}$. Under this assumption, argue that $\mathsf{SAT} \in \mathsf{RP}$.

5. **NP-hardness of approximation**
   Recall that a Minimization problem is efficiently α-approximable (for some $\alpha \geq 1$) if there is a polytime algorithm that finds an approximate solution whose value $APPROX$ satisfies: $OPT \leq APPROX \leq \alpha \cdot OPT$, where OPT is the value of an optimal solution.

   Consider the TSP problem:

   > Given a weighted complete graph $G = (V, E)$ on n vertices, with positive integer weights (in binary) on its edges $w : E \to \mathbb{Z}^+$, find the cost of a minimal-cost tour (where a tour is a Hamiltonian cycle in G and its cost is the sum of edge weights for the edges in the cycle).

   Show that, for every polynomial $\alpha(n) = n^c$, for $c \geq 0$, this problem is NP-hard to $\alpha(n)$-approximate. That is, show that if for some $\alpha(n) = n^c$, there is a deterministic polytime $\alpha(n)$-approximation algorithm for TSP on n-vertex graphs, then $\mathsf{P} = \mathsf{NP}$.