CS 281 – Homework 6

This assignment is due on **Friday, June 3** at the beginning of class.

Definition 1. For all $k \geq 1$, $L \in \Sigma_k$ if and only if there are a polynomial $q(n)$ and a relation $R(x, y_1, \ldots, y_k)$ in P such that

$$x \in L \Leftrightarrow \exists y_1 \forall y_2 \ldots Q_k y_k R(x, y_1, \ldots, y_k)$$

where the quantifies alternate ($Q_k = \forall$ if k is even, and $Q_k = \exists$ if k is odd) and $y_1, \ldots, y_k$ range over words of length $\leq q(|x|)$. Dually, $L \in \Pi_k$ if and only if

$$x \in L \Leftrightarrow \forall y_1 \exists y_2 \ldots Q'_k y_k R'(x, y_1, \ldots, y_k)$$

With this definition, notice that, you can view a series of alternating quantifiers as a Σ or Π machine, whichever is appropriate.

1. The following are equivalent for all $k \geq 1$: (10 pts)
   - $\Sigma_k = \Sigma_{k+1}$
   - $\Pi_k = \Pi_{k+1}$
   - $\Sigma_k = \Pi_k$
2. If for some $k \geq 1, \Sigma_k = \Pi_k$, then for all $j \geq k, \Sigma_j = \Pi_j = \Sigma_k$. Hint: first show that $\Sigma_k = \Pi_{k+1}$ and $\Pi_k = \Sigma_{k+1}$. (10 pts)
3. Problem 11.2 (20 pts)
   For each of these problems (1-4) give an algorithm for membership in the complexity class you suggest from classes: $\mathcal{P}$, $\mathcal{NP}$, *co*-$\mathcal{NP}$, $\mathcal{PS}$, $\mathcal{NPS}$, and recursive.
   (1) SP (Shortest Paths): given a weighted, undirected graph with nonnegetive integer edge weights, given two nodes in that graph, and given an integer limit k, determine whether the length of the shortest path between the nodes is k or less.
   (2) WHP (Weighted Hamilton Paths): given a weighted, undirected graph with non-negative integer edge weights, and given an integer limit k, determine whether the length of the shortest Hamilton path in the graph is k or less.
   (3) TAUT (Tautologies): given a propositional boolean formula, determine whether it is true for all possible truth assignments to its variables.
   (4) QBF (Quantified Boolean Formulas): given a boolean formula with quantifiers for-all and there-exists, such that there are no free variables, determine whether the formula is true.

   This algorithm will be a proof of upper bound, because it shows that the problem belongs to the proposed class. Also by placing them in the correct class, you claim a certain lower bound under the given assumptions. However, you do not have to prove your claim. In the statement of the first two problems, the word "length" should be replaced with "weight".