Complexity Theory 2012
(Home work 1)

June 12, 2012

- Due on Monday, June 26, before 10 a.m.
- Late home works will not be accepted.
- Please give precise arguments for all statements that you write.
- Please do not hesitate to contact me if you do not understand the problems.
- Collaboration is discouraged, but not prohibited. It is recommended that you try to solve the problems on your own. You can discuss the questions with your colleagues but you should not copy solutions. Always write down your own answers. If copying is detected that may immediately lead to a grade less than 7. (**This would be followed strictly**)
- Credits would be given to partial solutions also.
- The answers should be typed or written clearly and a hard copy is to be submitted.

1. **Warm up problems**.

   (a) If $L_1, L_2 \in \mathbf{NP}$, is $L_1 \cup L_2 \in \mathbf{NP}$?

   (b) If $L_1, L_2 \in \mathbf{NP}$, is $L_1 \cap L_2 \in \mathbf{NP}$?

   (c) If $L_1 \leq_p L_2$, is $L_2 \leq_p L_1$?

   (d) Prove that $P \subseteq \mathbf{NP} \cap \mathbf{coNP}$?

   (e) Prove that if $\mathbf{P} = \mathbf{NP}$, then $\mathbf{NP} = \mathbf{coNP}$.

   (f) For languages L_1, L_2, let us define $L_1 \oplus L_2$ as

   $$L_1 \oplus L_2 = \{x : x \text{ is in exactly one of } L_1, L_2\} = (L_1 \setminus L_2) \cup (L_2 \setminus L_1).$$

   Suppose $L_1, L_2 \in \mathbf{NP} \cap \mathbf{coNP}$, then show that $L_1 \oplus L_2 \in \mathbf{NP} \cap \mathbf{coNP}$

2. Let $\mathsf{HALT} = \{(\alpha, x) : M_\alpha \text{ halts on input } x\}$. Recall that M_α denotes the Turing machine represented by the string α. Show that HALT is **NP** hard. Is HALT, **NP** complete?

3. Let Pol be a language defined as

$$\mathsf{Pol} = \{< p >: \ p \text{ is a polynomial in any number of variables with integer roots}\}.$$

   Show that Pol is **NP** hard. Is it **NP** complete?

4. Given an undirected graph $G = (V, E)$, a coloring of G means to assign colors to the vertices of G such that no adjacent vertices gets the same color. Thus, a graph coloring is a function $c : V \to \mathsf{COLORS}$, where COLORS is a finite set whose elements are called colors, the function c should have the property that for every edge $(u, v) \in E$, $c(u) \neq c(v)$. G is called k colorable if it has a coloring with k colors, i.e, when k is a number we define

$$k\mathsf{COL} = \{G : G \text{ is } k \text{ colorable}\}$$

   It is known that $2\mathsf{COL} \in \mathbf{P}$ and $3\mathsf{COL}$ is **NP** complete. Assuming $3\mathsf{COL}$ is **NP** complete prove that $4\mathsf{COL}$ is **NP** complete.

5. Consider the language $\mathsf{MULTSAT}$ defined as

$$\mathsf{MULTSAT} = \{< \phi >: \phi \text{ is a CNF with at least 7 satisfying assignments}\}.$$

   Show that $\mathsf{MULTSAT}$ is **NP** complete.

6. Suppose there is a polynomial time Turing machine which decides the language

$$\mathsf{CLIQUE} = \{< G, k >: \text{the graph } G \text{ has a clique of size } k\}.$$

   Now, define the problem $\mathsf{sCLIQUE}$ as follows:

   Given an undirected graph $G = (V, E)$, find a clique of maximum size of G

   Show that there is a polynomial time procedure to solve $\mathsf{sCLIQUE}$.