Tutorial 10

Exercise 1 (compulsory)

Below are some definitions of the class NP. Which ones are correct?

1. NP is the class of languages which have polynomial time verifiers.
2. NP is the class of languages that cannot be decided in polynomial time using a deterministic Turing machine.
3. NP is the class of languages that have nondeterministic verifiers.
4. NP is the class of languages that can be decided in polynomial time on a nondeterministic Turing machine.

Exercise 2 (compulsory)

Consider the following decision problem, also know as *bin packing*. Given a finite set S of natural numbers, a number k of available bins, and a maximum capacity M of the bins, determine whether the items can be partitioned into $S_1, \ldots, S_k$ such that $\bigcup_{1\leq i\leq k} S_i = S$ and the sum of the items in each bin does not exceed the maximum capacity M, i.e., $\sum S_i \leq M$ for all i, $1 \leq i \leq k$.

1. Define a language $BINPACK$ for the above mentioned decision problem.
2. Argue that $BINPACK \in$ NP by constructing a polynomial time nondeterministic TM deciding it.
3. Argue that $BINPACK \in$ NP by constructing a polynomial time verifier for $BINPACK$.

Exercise 3 (compulsory)

Let $PATH \stackrel{\text{def}}{=} \{\langle G, s, t\rangle \mid G$ is a graph, s and t two nodes in the graph and there is a path from s to t $\}$.

Each of the following two algorithms claim to decide $PATH$ in nondeterministic polynomial time (and hence conclude that $PATH$ belongs to NP). However, they both contain an error. Find it and explain what is wrong.

1. "On input $\langle G, s, t\rangle$:
   1. Nondeterministically select a number $k \in \mathbb{N}$. (The length of a path from s to t.)
   2. Nondeterministically select k nodes in the graph G.
   3. Verify whether the first node is s, the last one is t, and they are all connected by edges.
   4. If yes, then accept, else reject."
2. "On input $\langle G, s, t\rangle$ of length n:
   1. Nondeterministically select a number $k \leq 2^n$.
      (Note that k written in binary is of polynomial length w.r.t. to n.)
   2. Nondeterministically select k nodes in the graph G.
   3. Verify whether the first node is s, the last one is t, and they are all connected by edges.
   4. If yes, then accept, else reject."

Because you found an error in both of the algorithms, can you so conclude that $PATH \notin$ NP?

Exercise 4 (compulsory)

Prove that co-NP $\subseteq$ EXPTIME.

Exercise 5 (compulsory)

Prove that $\leq_P$ is transitive. In other words show that if $L_1 \leq_P L_2$ and $L_2 \leq_P L_3$ then $L_1 \leq_P L_3$. Do not forget to carry on a complexity analysis of the construction.

Exercise 6 (optional, but interesting)

Consider the decision problem

> Given an undirected graph G, is it the case that G has a clique of size 4?

1. Express this problem as the language *FOURCLIQUE*.
2. Prove that *FOURCLIQUE* $\in$ NP.
3. Prove that *FOURCLIQUE* $\in$ P.
4. Do we now know that *CLIQUE* $\in$ P? Justify your answer.