Exercise 6 - Computational Models

1. For the following decision problems, assuming $P \neq NP$, determine whether they are in P or in NPC. Prove your answer.

   (a) Input: sets $A_1, ..., A_n$, and a number k.
   Question: do there exist k mutually disjoint sets $A_{i_1}, ..., A_{i_k}$?

   (b) Input: sets $A_1, ..., A_n$, and a number k.
   Question: do there exist k sets $A_{i_1}, ..., A_{i_k}$ such that $A_{i_j} \cap A_{i_l} \neq \emptyset$ for every $j \neq l$?

   (c) Input: a 3CNF formula ψ.
   Question: does there exist an assignment that satisfies ψ and gives $True$ for exactly 10 variables?

   (d) Input: a 3CNF formula ψ with even number of variables.
   Question: does there exist an assignment that satisfies ψ and gives $True$ for exactly one half of the variables?

   (e) Input: directed graph G, a number k.
   Question: does there exist a simple path in G of length $\geq k$?

2. Determine whether the following claims are true. false or equivalent to an open problem:

   (a) For every nontrivial $L_1, L_2 \in P$, if $L_1 \leq_m L_2$ then $L_1 \leq_p L_2$.

   (b) For every nontrivial $L_1, L_2 \in NP$, if $L_1 \leq_m L_2$ then $L_1 \leq_p L_2$.

   (c) There exists a language in RE that is complete w.r.t polynomial-time reductions.

   (d) $L = \{xx | x \in \{0,1\}^*\}$ is NPC.

   (e) If there exists a deterministic TM that decides SAT in time $n^{O(logn)}$ then every $L \in NP$ is decidable by a deterministic TM in time $n^{O(logn)}$.

3. We define $L^* = \{w_1 w_2 ... w_k | 0 \leq k, \forall i \leq k \; w_i \in L\}$. Prove:

   (a) If $L \in NP$ then $L^* \in NP$.

(b) If $L \in P$ then $L^* \in P$.

4. We say that a polynomial reduction f is a *shrinking reduction* if there exists n_0 such that for every $x \in \Sigma^*$ such that $n_o \leq |x|, |f(x)| \leq |x| - 1$. Assuming $P \neq NP$, prove/disprove:

   (a) For every two nontrivial languages $A, B \in P$ there exists a shrinking reduction from A to B.

   (b) For every two nontrivial languages $A, B \in NPC$ there exists a shrinking reduction from A to B.

5. Prove Claim 1, appeared Claim 15 in Lecture 11, using Claim 1, appeared as Claim 18 in Lecture 11. (See the proof guidelines appears in Slide 21.)

   Claim 1. *Let $w = xyz$, and assume that in $M(w)$ has identical crossing sequences at $|x|$ and $|x + y|$, then $M(w)$ and $M(w' = (x, y^2, z))$ halt in the same state.*

   Claim 2. *Let $\mathsf{conf}_{\mathsf{z,t}}(\mathsf{i}, \mathsf{k})$ be the augmented configuration of $M(z)$ after t steps at $[i, k]$. For any $t \in N$, it holds that*

   (a) $\exists t_1$ *such* $\mathsf{conf}_{\mathsf{w',t}}(1, |\mathsf{x} + \mathsf{y}|) = \mathsf{conf}_{\mathsf{w,t_1}}(1, |\mathsf{x} + \mathsf{y}|)$

   (b) $\exists t_2$ *such* $\mathsf{conf}_{\mathsf{w',t}}(|\mathsf{x} + \mathsf{y}| + 1, \inf) = \mathsf{conf}_{\mathsf{w,t_2}}(|\mathsf{x}| + 1, \inf)$

6. Consider the following language: $L = \{\#1^n\#x_1...x_n|\exists i, j\ s.t : x_i = x_j\}$. A possible way to implement a Turing Machine that accepts the language is as follows: "*Choose* (non-deterministically) two indices i and j $(0 < i, j \leq n)$ and write them on a second tape. Now check (using a deterministic TM) if (1) $i \neq j$ and (2)$x_i = x_j$".
   In this exercise we will $formally$ implement the first part: on a $two-taped, non-deterministic$ Turing Machine with input $\#1^n\#(n > 1)$, choose non-deterministically i and j such that $0 < i, j \leq n$ and write on the second tape $1^i\#1^j$ and accept. If this helps you, you may use a model that allows the head to stay put as well as moving right and left.