Tutorial 4

Exercise 1: Recall the definition of class PTIME. Give examples of at least three problems from PTIME. For each of these problems:

a) Specify precisely what is an input and what is a question.

b) Give examples of instances, for which the answer is YES, and examples of instances, for which that answer is NO.

c) Show that the problem is actually in PTIME, i.e., describe a polynomial algorithm solving the given problém.

Exercise 2: Show that the following problem belongs to PTIME:

> INPUT: Natural numbers a, b, c, m, where $m > 1$, represented in binary.
>
> QUESTION: Is $a^b \equiv c \pmod{m}$?

Exercise 3: A permutation on a set $S = \{1, 2, \ldots, k\}$ an arbitrary bijective map $p : S \to S$. Let p^t, where $t \in \mathbb{N}$, be the function that is obtained by composing function p with itself t times.

Show that the following problem is in PTIME:

> INPUT: A pair of permutations p and q on set $\{1, 2, \ldots, k\}$ and a natural number t represented in binary.
>
> QUESTION: Is $p^t = q$?

Exercise 4: Define precisely the notion of ***polynomial reduction*** as a special case of a reduction between problems discussed before.

Describe in detail what should be done if we want to show a polynomial reduction from problem HC (Hamiltonian circuit in a directed graph) to problem HK (Hamiltonian circuit in an undirected graph).

Then show this reduction.

Exercise 5: Explain in detail the notion of "NP-complete problem".

Define problems SAT, 3-SAT, HC, HK, 3-CG, and IS (with examples of positive and negative instances). These problems are NP-complete. For each of these problems, describe an algorithm showing that the given problem is in NP.

Exercise 6: Recall the definitions of the clique problem (CLIQUE) and the vertex cover problem (VC).

Show how to reduce the independent set problem (IS) by polynomial reductions to these problems.

Exercise 7: Consider the following problem (one of the often given examples of NP-complete problems).

> NAME: TSP (*travelling salesman problem (*YES/NO *version)*)
>
> INPUT: A set of "cities" $\{1, 2, \ldots, n\}$, natural numbers ("distances") d_{ij} ($i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, n$), and number ℓ („limit").
>
> QUESTION: Is there a cycle going through all cities of length at most ℓ, i.e., is there a permutation $(i_1, i_2, \ldots, i_n)$ of set $\{1, 2, \ldots, n\}$ such that
>
> $$d(i_1, i_2) + d(i_2, i_3) + \cdots + d(i_{n-1}, i_n) + d(i_n, i_1) \leq \ell\,?$$

This is a decision (or YES/NO) version of an optimization problem. Derive at first this optimization problem (i.e., what is its input and what is the corresponding output).

Then show some small (but not completely trivial) instance of the given problem TSP, for which the answer is YES, and some instance, for which the answer is NO.

Then show that TSP in in NP (by describing a nondeterministic algorithm).

Finally, try to propose a proof of NP-hardness of TSP.

Hint: You can use the fact that the hamiltonian circuit problem on undirected graphs (HK) is NP-complete.

Exercise 8: Try to describe in detail how 3-SAT can be reduced by a polynomial reduction to hamiltonian path problem.

This proof can be found for example in book M. Sipser: *Introduction to the Theory of Computation*, 2nd edition, Course Technology, 2006, in Section 7.5 (*Additional NP-complete Problems*).