Albert-Ludwigs-Universität, Inst. für Informatik
Prof. Dr. Fabian Kuhn
M. Ahmadi, P. Bamberger, H. Ghodselahi, Y. Maus, P. Schneider

Theoretical Computer Science - Bridging Course
Summer Term 2017
Exercise Sheet 8

Hand in (electronically or hard copy) by 12:15 pm, July 10, 2017

Exercise 1: $\mathcal{NP}$ and Star Operation (5 points)

Show that $\mathcal{NP}$ is closed under the star operation.

*Remark 1: Let A be a language. The operation **star** $(\cdot^*)$ is defined as follows:*

$$A^* = \{x_1x_2\dots x_k \mid k \geq 0 \text{ and each } x_i \in A \text{ where } 0 \leq i \leq k\}.$$

*Remark 2: A collection of objects is **closed** under some operation if applying that operation (a finite number of times) to members of the collection returns an object still in the collection.*

Exercise 2: The class $\mathcal{NPC}$ (8 points)

Let L_1, L_2 be languages (problems) over alphabets Σ_1, Σ_2. Then $L_1 \leq_p L_2$ (L_1 is polynomially reducible to L_2), iff a function $f : \Sigma_1^* \to \Sigma_2^*$ exists, that can be calculated in polynomial time and

$$\forall s \in \Sigma_1 : s \in L_1 \iff f(s) \in L_2.$$

Language L is called $\mathcal{NP}$-hard, if *all* languages $L' \in \mathcal{NP}$ are polynomially reducible to L, i.e.

$$L \ \mathcal{NP}\text{-hard} \iff \forall L' \in \mathcal{NP} : L' \leq_p L.$$

The reduction relation '$\leq_p$' is transitive ($L_1 \leq_p L_2$ and $L_2 \leq_p L_3 \Rightarrow L_1 \leq_p L_3$). Therefore, in order to show that L is $\mathcal{NP}$-hard, it suffices to reduce a known $\mathcal{NP}$-hard problem $\tilde{L}$ to L, i.e. $\tilde{L} \leq_p L$. Finally a language is called $\mathcal{NP}$-complete ($\Leftrightarrow$: $L \in \mathcal{NPC}$), if

1. $L \in \mathcal{NP}$ and
2. L is $\mathcal{NP}$-hard.

Show $\textsc{HittingSet} := \{\langle \mathcal{U}, S, k\rangle \mid \text{universe } \mathcal{U} \text{ has subset } H, |H| \leq k \text{ that } \textbf{hits} \text{ all sets in } S \subseteq 2^{\mathcal{U}}\} \in \mathcal{NPC}$.[1]

Use that $\textsc{VertexCover} := \{\langle G, k\rangle \mid \text{Graph } G \text{ has a } \textit{vertex cover} \text{ of size at most } k\} \in \mathcal{NPC}$.

*Remark: A **hitting set** $H \subseteq \mathcal{U}$ for a given universe $\mathcal{U}$ (which is a **finite** set) and a set $S = \{S_1, S_2, \dots, S_m\}$ of subsets $S_i \subseteq \mathcal{U}$, fulfills the property $H \cap S_i \neq \emptyset$ for $1 \leq i \leq m$ (H 'hits' at least one element of every S_i).*
*A **vertex cover** is a subset $V' \subseteq V$ of nodes of $G = (V, E)$ such that every edge of G is adjacent to a node in the subset.*

Hint: For the poly. transformation ($\leq_p$) you have to describe an algorithm (with poly. run-time!) that transforms an instance $\langle G, k\rangle$ of $\textsc{VertexCover}$ *into an instance $\langle \mathcal{U}, S, k\rangle$ of* $\textsc{HittingSet}$*, s.t. a vertex cover of size $\leq k$ in G becomes a hitting set of $\mathcal{U}$ of size $\leq k$ for S and vice versa(!).*

[1] The power set $2^{\mathcal{U}}$ of some ground set $\mathcal{U}$ is the set of *all subsets* of $\mathcal{U}$. So $S \subseteq 2^{\mathcal{U}}$ is a collection of subsets of $\mathcal{U}$.

Exercise 3: Complexity Classes: Big Picture (2+3+2 points)

(a) Why is $\mathcal{P} \subseteq \mathcal{NP}$?

(b) Show that $\mathcal{P} \cap \mathcal{NPC} = \emptyset$ if $\mathcal{P} \neq \mathcal{NP}$.
Hint: Assume that there exists a $L \in \mathcal{P} \cap \mathcal{NPC}$ and derive a contradiction to $\mathcal{P} \neq \mathcal{NP}$.

(c) Give a Venn Diagram showing the sets $\mathcal{P}, \mathcal{NP}, \mathcal{NPC}$ for both cases $\mathcal{P} \neq \mathcal{NP}$ and $\mathcal{P} = \mathcal{NP}$.
Remark: Use the results of (a) and (b) even if you did not succeed in proving those.