Theory of Computer Science

M. Helmert
T. Keller
Spring Term 2017

University of Basel
Computer Science

Exercise Sheet 11

Due: Sunday, May 28, 2017

Note: Submissions that are exclusively created with LaTeX will receive a bonus mark. Please submit only the resulting PDF file (or a printout of this file).

Exercise 11.1 (1.5+1.5+1 marks)

Consider the decision problem CLIQUE:

- *Given:* undirected graph $G = \langle V, E \rangle$, number $K \in \mathbb{N}_0$
- *Question:* Does G contain a clique of size K or more, i.e., a set of nodes $C \subseteq V$ with $|C| \geq K$ and $\{u, v\} \in E$ for all $u, v \in C$ with $u \neq v$?

(a) Specify a non-deterministic algorithm for CLIQUE, whose runtime is limited by a polynomial in $|V| + |E|$. Explain why the algorithm's runtime is polynomial.

(b) Specify a deterministic algorithm for CLIQUE.

(c) Estimate the runtime of your algorithm from part (b) in O-Notation.

You can use any common programming concepts in your answer. You do *not* have to use the restricted syntax of WHILE-programs or similar languages. High-level pseudo code is sufficient as long as it can be easily seen that each step runs in polynomial time. Use the GUESS statements from the lecture for non-deterministic statements.

Exercise 11.2 (1+1+1.5+1.5 marks)

Prove or refute the following statements. In all cases, specify a short proof (2–3 sentences are sufficient).

(a) Let X be an NP-hard problem and Y a problem with $X \leq_{\text{p}} Y$. Then Y is NP-hard.

(b) Let X be an NP-hard problem. If there is a deterministic polynomial algorithm for X, then there also is a deterministic polynomial algorithm for DIRHAMILTONCYCLE.

(c) There are NP-complete problems X and Y where there is a deterministic polynomial algorithm for X but not for Y.

(d) Let $Y \subseteq \Sigma^*$ be any problem with $Y \neq \emptyset$ and $Y \neq \Sigma^*$. Then $X \leq_{\text{p}} Y$ holds for all $X \in \text{P}$.

Exercise 11.3 (2+1 marks)

A *Hamilton path* is defined analogously to a Hamilton cycle (see chapter E1) with the only difference that we look for a simple path instead of a cycle. More formally: a Hamilton path in a directed graph $\langle V, E \rangle$ is a sequence of vertices $\pi = \langle v_1, \ldots, v_n \rangle$ that defines a path ($\langle v_i, v_{i+1} \rangle \in E$ for all $1 \leq i < n$) and contains every vertex in the graph exactly once.

Consider the decision problem DIRHAMILTONPATH:

- *Given:* directed graph $G = \langle V, E \rangle$
- *Question:* Does G contain a Hamilton path?

(a) Prove that DIRHAMILTONPATH is NP-hard. You can use without proof that DIRHAMILTONCYCLE is NP-complete.

(b) Is DIRHAMILTONPATH NP-complete? Justify your answer.