TU Dresden, Fakultät Informatik
Stephan Mennicke, Sergei Obiedkov

Winter Term 2025/2026

Complexity Theory

Exercise 10: Randomised Computation

20 January 2026

Exercise 10.1. Show that **MajSat** is in PP.

$$\textbf{MajSat} = \{\varphi \mid \varphi \text{ is some propositional-logic formula that is satisfied by more than half of its assignments}\}$$

Exercise 10.2. Alice and Bob have a string of n bits each. They want to find out if their strings are the same. They exchange messages following a certain protocol. A person's message may depend on the string one has and on all the messages exchanged so far. Show that any deterministic protocol requires Alice and Bob to exchange at least n bits to be able to reach a correct conclusion.

Next, devise a randomized protocol that requires only $O(k)$ bits to reach a correct conclusion with probability at least $1-2^{-k}$. You may assume that Alice and Bob share access to an infinite tape with random bits, so that, if they need a random bit string, they can use the same string without having to communicate it to each other. Hint: use fingerprinting.

Exercise 10.3. Suppose that $\mathrm{NP} \subseteq \mathrm{BPP}$. Show that, in this case, there is an algorithm that runs in expected polynomial time and, given a satisfiable Boolean formula φ, finds a satisfying assignment for φ.

Exercise 10.4. Show $\mathrm{BPP} = \mathrm{coBPP}$.

Exercise 10.5. Let φ be a CNF in which each clause contains an arbitrary positive number of literals. Assume that all literals within a clause are distinct and that there is no variable x such that φ contains both clauses (x) and $(\neg x)$. Design a polynomial-time algorithm that, given such a formula φ with m clauses, computes a variable assignment that, in expectation, satisfies at least $0.6m$ clauses.

Exercise 10.6. Suppose that there is a polynomial-time randomized algorithm that, for any input graph G and natural number k, with probability at least 2/3, correctly reports whether G has a clique of size k. Assume that, on a graph with n vertices, the algorithm makes at most $\log_2 n$ randomized steps, each time choosing uniformly at random between two options; all other steps are deterministic.

Prove that, in this case, $\mathrm{P} = \mathrm{NP}$.

Exercise 10.7. Show that $\mathrm{DSPACE}(n^{\log n}) \not\subseteq \mathrm{BPP}$.