

# Übung zur Vorlesung BERECHENBARKEIT UND KOMPLEXITÄT

## Lernhilfe zum Thema Komplexität

---

Die Klausur und die Präsenzübung bestehen zu einem großen Teil aus Wissens- und einfachen Verständnisfragen. Der nachfolgende Aufgabenkatalog soll dazu dienen, das eigene Wissen zu kontrollieren und ein Gefühl für mögliche Aufgabenstellen zu bekommen.

Einige dieser Fragen werden sich sicherlich auch in den Klausuren wiederfinden.

1. Wie ist die worst-case Laufzeit eines Algorithmus definiert?
2. Definieren Sie die Komplexitätsklasse  $P$ .
3. Was ist eine nicht-deterministische Turingmaschine (NTM)? Wie ist ihr Akzeptanzverhalten und ihre Laufzeit definiert?
4. Definieren Sie die Komplexitätsklasse  $NP$  mit Hilfe von NTMs einerseits und Polynomialzeitverifizierern andererseits. Zeigen Sie die Äquivalenz der beiden Definitionen.
5. Geben Sie die Optimierungs- und Entscheidungsvarianten der folgenden Probleme an: Clique, KP, BPP, TSP.
6. Zeigen Sie für die obigen Probleme, dass sich aus polynomiellen Algorithmen für die Entscheidungsvarianten polynomielle Algorithmen für die Optimierungsvarianten folgern lassen, und umgekehrt.
7. Geben Sie die Simulation einer NTM mit polynomieller worst-case Laufzeit durch eine TM an. Wie hoch ist die Laufzeit der resultierenden TM?
8. Nennen Sie den Unterschied zwischen  $\leq$  und  $\leq_p$ .
9. Wann heißt ein Problem  $NP$ -hart bzw.  $NP$ -vollständig?
10. Beschreiben Sie kurz die Aussage und die Beweisidee des Satzes von Cook und Levin. (Wiederholen Sie sorgfältig den Beweis.)
11. Wiederholen Sie sämtliche polynomielle Reduktionen aus der Vorlesung und den Übungen.
12. Erläutern Sie das Verhältnis der Komplexitätsklassen  $P$ ,  $NP$  und  $NPC$  zueinander unter der Annahme  $NP \neq P$ . ( $NPC$  bezeichnet die Klasse der  $NP$ -vollständigen Probleme.)
13. Definieren Sie die starke und die schwache  $NP$ -Härte.
14. Was versteht man unter pseudopolynomieller Laufzeit?
15. Was ist ein PTAS? Was ist ein FPTAS?
16. Geben Sie ein FPTAS für das Rucksackproblem an.
17. Definieren Sie die Approximationsgüte eines Algorithmus.