Introduction to Algorithm Theory

Exercises with Dino's 6th lecture

Problems 40, 41 and 43 in the Compendium

Discuss the following questions:

1. We organize problems into complexity classes to understand what solution strategies are suitable. What solution strategies are appropriate for NP-complete problems?
2. What is a polynomial-time approximation algorithm? Do all NP-complete problems have polynomial-time approximation algorithms?
3. Why do we talk about random graphs when discussing the average-case complexity?
4. What are 0-1 laws? Why are they relevant for us?
5. Outline the three-phase algorithm for Hamiltonicity that was discussed in class. In what way does this algorithm suggest us how to solve problems 'well on average' in general?
6. Can we we create polynomial-time randomized algorithms for problems that are not in NP (i.e. do not have a 'short certificate of membership')?
7. What is simulated annealing? In what application is it used?
8. Can we solve NP-complete problems efficiently by using a parallel computer?
9. What sort of problems can be solved radically more efficiently when we use a parallel machine instead of a sequential one?
10. What sort of problems are in the class NC?