

Study Problems for NP Completeness

CISC 365, Fall 2010

Focus your studying on the five definitions:

1. Decision Problem.
Be familiar with the decision problems we discussed in lecture: TSP (Traveling Salesman problem) and Satisfiability.
Be able to formulate a regular problem as a corresponding decision problem. Example done in class: regular TSP and decision TSP. (Also, in part 4. below: be able to show that $\text{DecisionTSP} \leq_p \text{RegularTSP}$). Another example is the independent set problem: start of textbook problem 34-1a, page 1102.
2. Class P
Be able to prove that a problem is in class P: state a polynomial time algorithm.
3. Class NP
Be able to prove that a problem is in class NP: state what a guessed solution (a certificate) consists of, and state how a guessed solution can be checked in polynomial time. Examples discussed in class: TSP and Satisfiability. Another example is graph isomorphism, textbook problem 34.2-1, page 1065.
4. Reduction
Be able to judge the correctness of a given reduction, and be able to construct simple reductions on your own (example: $\text{DecisionTSP} \leq_p \text{RegularTSP}$).
Understand the implications of $A \leq_p B$, and be able to justify the following two statements.
If we know that $A \leq_p B$:
 - (i) if there is a polynomial algorithm for B \Rightarrow there exists a polynomial algorithm for A
 - (ii) if A is NP-complete (and B is in class NP) \Rightarrow B is NP-complete
5. Class NP-Complete. The hardest problems in class NP.
Be able to set up an NP-completeness proof: state what has to be proven (as in “find a reduction from blah to blah, in other words, a polynomial-time translation...”). Various examples are given in the textbook, including the start of textbook problem 34-1a, page 1102: set up this proof by stating what needs to be proven, but you don’t actually have to carry out the proof.
Understand the proof (posted on the course website) that the Set Intersection problem is NP-complete.