

CSE 200
Computability and Complexity
Homework 3
The class NP and beyond, Completeness, and
Reductions
Polynomial-time Hierarchy
Space Complexity
Due May 26

May 11, 2010

Give proofs for each problem. Proofs can be high-level, but be precise. You may use without giving a proof any result proved in class or in the textbook. In particular, to prove NP-completeness, it suffices to give a reduction from any of the NP-complete problems from the text or from class. However, you must show your reduction is valid, by showing the equivalence of the constructed instance and the original.

NP and $co - NP$ Consider the problem Greater Maximum Independent Set: Given G_1 and G_2 , is the maximum independent set for G_1 strictly larger than that for G_2 ? Show that this problem is in NP if and only if it is in $Co - NP$ if and only if $NP = co - NP$.

NP-Completeness We say that NP complete problems are the “hardest” in NP , but intuitively that means they are the “least likely to be easy” not that they have the greatest worst-case complexity. To illustrate this, prove that for every $k > 0$ there is an $NP - complete$ language L so that $L \in TIME(2^{n^{1/k}})$.

Polynomial-time hierarchy Assume that $NP \subseteq TIME(n^{O(\log n)})$. Prove that $\Sigma_i \subseteq TIME(n^{O((\log n)^{2^i - 1})})$.

Space complexity Let $F \in FP$ be a function so that $|F(x)| = |x|$ for every x . Let F^k be F composed with itself k times, so $F^1(x) = F(x)$, and $F^{k+1}(x) = F(F^k(x))$. A *fixed point* for F on x is a value of k so that $F^{k+1}(x) = F^k(x)$. Let $F - FIX = \{x \mid \text{there is some fixed point } k \text{ of } F \text{ on } x\}$. Prove that $P = PSPACE$ if and only if $F - FIX \in P$ for every such $F \in FP$.

Sudoku experiment We started looking at reducing sudoku problems to *SAT* last assignment.

The *sudoku* problem of size n is as follows. The input is an $n^2 \times n^2$ matrix M whose entries are either “blank” or an integer between 1 and n^2 . A solution fills in the blank spaces with integers between 1 and n^2 . The following constraints must be met: Each integer from 1 to n^2 appears exactly once in each row, in each column, and in each $n \times n$ sub-matrix of the form $M[jn + 1 \dots (j + 1)n][in + 1 \dots (i + 1)n]$ for each $0 \leq i, j \leq n - 1$. The problem is to find any solution meeting the constraints, or return “no solution possible” if there is no such solution.

Last assignment, you gave at least two different ways to reduce the Sudoku problem to *CNF - SAT*.

Try solving sudoku problems by combining the above reductions with a complete SAT solver, such as Zchaff (download page, <http://www.princeton.edu/~chaff/zchaff/index2.html>)

Use as test inputs the puzzles of sizes 9 by 9, 16 by 16, and 25 by 25 from <http://www.doncasterwx.co.uk/Colinj/>

Be sure to credit both websites.

The experiment should return the following information: For each size, and each difficulty rating, how much time did the SAT solver take using different reductions? Which reduction is best? Did the difficulty ratings reflect in the time taken by these solvers?

(Note: Be careful not to use up too much computer time. Don't leave programs running unsupervised too long. Depending on your algorithm and reduction, you may find even very small sizes take huge amounts of time.)