Institut for Matematik og Datalogi
Syddansk Universitet

April 24, 2014
JFB

DM508 – Algorithms and Complexity – 2014
Lecture 5

Lecture, April 23

We continued with NP-Completeness, looking more at reductions (section 34.3 in the textbook) and showing that 3-SAT and CLIQUE, are NP-Complete. The latter two reductions are in sections 34.4 and 34.5 in the textbook.

Lecture, April 29

We will cover Cook's Theorem, proving that SATISFIABILITY from the section by Papadimitriou and Steiglitz from the course notes. We may also show that VERTEX COVER and INDEPENDENT SET are NP-Compete. This is in section 34.5 in the textbook.

Lecture, May 6

We will show that HAMILTONIAN CIRCUIT, SUBSET SUM, VERTEX COVER and INDEPENDENT SET are NP-Compete. This is in section 34.5 in the textbook.

Problems to be discussed in U155 on May 8, 14–16

Do problems:

1. 34.2-5, 34.2-10.
2. 34.3-2, 34.3-6.
3. The following argument is incorrect. Find the most important error.

   Consider the following algorithm:

   Input: $n \in \mathbb{N}$

   for $i = 2$ to $n-1$ **do**
   check if i divides n
   if it does **then** output i
   endfor
   output -1 if no output yet

Checking if i divides n can be done in time $O(\log n)$ via binary search for an integer k such that $n = i \cdot k$.

Thus, the total running time is $O(n \cdot \log n)$ in the worst case. Since $O(n \cdot \log n) \subset O(n^2)$, and n^2 is a polynomial, this algorithm runs in polynomial time. Thus, we have an efficient algorithm for factoring, $O(n \cdot \log n)$, so we can break RSA, a famous cryptosystem which is believed to be secure.

Assignment due Tuesday, May 13, 8:15

Note that this is part of your exam project, so it must be approved in order for you to take the exam in June, and you may not work with or get help from others not in your group (though you may talk with Christian Kudahl or myself). You may work in groups of two or three. You may write your solutions in English or Danish, but write very neatly if you do it by hand. Submit the assignment via Blackboard's "SDU Assignment" as one PDF file, with no Danish letters in the file name. Remember to keep a receipt. Turn in one assignment per group.

1. Consider the following game, which we will call "Cave Tunnels". Every time you start this game, you are given a new map of a cave system with n *r*ooms, some of which are connected by *t*unnels, plus a set of k tokens, which we will call *blockers*. Your goal is to *control* as many of the tunnels as possible. You can only control a tunnel by placing a blocker in both of the rooms the tunnel connects.

   (a) Prove that this problem is NP-hard, by defining a recognition version of the problem (a decision problem) which is NP-Complete, and proving that the problem you define is NP-Complete. Call the decision problem you define "Decide Cave Tunnels".

   (b) Show that if there is a polynomial time algorithm for Decide Cave Tunnels, then there is a polynomial time algorithm for the evaluation version of Cave Tunnels (find the maximum number of tunnels which can be controlled with k blockers).

   (c) Show that if you can find a polynomial time algorithm to solve Decide Cave Tunnels, then you can also find a polynomial time algorithm to place the blockers to control a maximum number of tunnels in the Cave Tunnels game.

2. Show that the following problem is NP-hard: Given two Boolean formulas, F_1 and F_2, is exactly one of them satisfiable?