Institut for Matematik og Datalogi
Syddansk Universitet

April 21, 2015
JFB

DM553 Lecture 12 — DM508 Lecture 3

Lecture, April 20

We began showing that 3-SAT is NP-Complete. To do this, we assume that CNF-SAT is NP-Complete. The proof presented combines the proofs in both the Sipser and the CLRS books. Then we covered the proof that SATISFIABILITY is NP-Complete, from section 7.4 in Sipser's textbook.

Lecture, April 22

We will show that CNF-SAT, CLIQUE, VERTEX COVER, INDEPENDENT SET, and SUBSET SUM are NP-Complete. See chapter 34 in CLRS.

Lecture, April 27

We will show that HAMILTONIAN CIRCUIT is NP-Complete. Then we will start on lower bounds from section 2.4 in the notes. (Part of this is also in section 8.1 of CLRS.)

Problems to be discussed in U142 on April 28

1. 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 $O(\log^3 n)$ via binary search for an integer k such that $n = i \cdot k$ (the multiplication can clearly be done in time $O(\log^2 n)$).

   Thus, the total running time is $O(n \cdot \log^3 n)$ in the worst case. Since $O(n \cdot \log^3 n) \subset O(n^2)$, and n^2 is a polynomial, this algorithm runs in polynomial time. Thus, we have an efficient algorithm for factoring, so we can break RSA, a famous cryptosystem which is believed to be secure.

2. 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.

3. In the CLRS textbook, do the following:

   - 34.2-5, 34.2-10.
   - 34.3-6.