Institut for Matematik og Datalogi
Syddansk Universitet

March 16, 2016
JFB

DM553 – Complexity and Computability – 2016
Lecture 10

Lecture, March 14

We finished chapter 5, skipping the last part of section 5.1, having to do with reductions via computation histories.

Lecture, March 17 in U140

We will start with a course evaluation of how the course has been so far. Then, we will begin on NP-Completeness, introducing definitions and showing that 3-SAT is NP-Complete. To do this, we assume that CNF-SAT is NP-Complete. The proof that 3-SAT is NP-Complete combines the proofs in both the Sipser and the CLRS books. The definition of time complexity classes is on page 279 in Sipser's textbook, and the definition of P is on page 286. The definitions of NP and NP-Complete are in sections 7.3 and 7.4 of Sipser's textbook. Note that some of this is also in chapter 34 in the CLRS book.

Lecture, March 29

We will cover the proof that SATISFIABILITY (actually CNF-SAT) is NP-Complete, from section 7.4 in Sipser's textbook. If there is time, we will do more reductions from chapter 34 in CLRS.

Problems to be discussed in U155 on April 1

In the CLRS textbook, do the following:

1. 34.1-3, 34.1-5, 34.2-3.
2. Suppose that there is a language L for which there is an algorithm that accepts any string $x \in L$ in polynomial time and rejects any $x \notin L$, but this algorithm runs in super-polynomial (more than polynomial) time if $x \notin L$. Argue that L can be decided in polynomial time.
3. 34.2-4 (skip Kleene star), 34.2-8.
4. 34.3-7 (34.3-6 has the definition of complete you need).