Lecture 10: NP-Completeness and the Cook-Levin Theorem

We say that L is polynomially reducible to L', written $L <_P L'$ if for some a and b there is a DTM such that for any input w: (i) the machine halts after $a||w||^b$ steps, and (ii) the string written when the DTM halts consists of a word f(w) using only non-blanks followed by an infinite string of blanks, and (iii) f(w) is in L' precisely if w is in L.

A language L is NP-hard if every language in NP is polynomial reducible to L. A language L is NP-complete if (i) it is in NP, and (ii) it is NP-hard.

These definitions extend naturally to decision problems, i.e. problems which have a yes-no answer. The decision problem is in P, in NP, or NP-complete precisely if the language encoding yes instances of the problem is.

We observed that Cook and Levin had independently proved that the SAT decision problem is NP-complete. We worked on developing a proof ourselves. Your text for COMP330 was Sipser's Introduction to The Theory of Computation. The 7th chapter of that text contains a proof of the Cook-Levin theorem.