COS522: Computational Complexity Fall 2011
Princeton University
Instructor: Sanjeev Arora

Lecture 1 notes.

1. The basic model (Turing machine) and why we use it –simple, elementary, able to simulate all other realistic models with polynomial slowdown.
2. Defn of P, NP. Philosophical musings about why these classes are fundamental and model independent (strong form of Church-Turing thesis).
3. Examples of NP problems.
4. NP-completeness. Defn, trivial NPC problem, nontrivial NPC problems. Sketch of Cook-Levin theorem. Main insight: computation is local!
5. Discussion goals of complexity theory and preview of topics to come.
6. One central goal: capture computational phenomena and classes of problems using a complexity class. Illustration: NP (captures phenomenon that often checking a solution is much easier than discovering it).
7. Other phenomena we will try to capture or formalize via complexity: memory requirement (space), time on a parallel computer (multiprocessor), size of straight line program, alternation, games between rational players, randomness, quantum effects, cryptography (creating a problem can be easier than solving it), derandomization, counting/enumeration, approximation.
8. Another example of interconnection: If factoring has no subexp. algorithm then randomized algorithms have quasipolynomial deterministic simulations. (Will require developing some machinery.)

   Reading: Arora-Barak Chapters 0—2.