Homework #5: P and NP
Due Date: 5pm Tuesday 4/26 (reading day)

Administrative: Our last regular lecture is Thursday 4/21. I have scheduled a final review session in our usual time and room, on Tuesday 4/26 (a “reading day”). Our final exam is 8am Wednesday 5/4, in our usual room. The exam will be about facts and theorems, not creative problem solving (like homeworks). Also, I’ve decided to let you prepare a single sheet of notes, to consult during the exam (as seen in the sample exam, 0112/final-s14.pdf).

This is our last homework, and it is a bit different. There is no pairing up, you must work on your own. You may of course use your textbook and my office hours, but no other people or resources. Finally, I expect each of you to make a half-hour appointment to see me, sometime between 4/25 and 5/5 (inclusive), to discuss this homework. I’ll ask you present one or more of your solutions at the whiteboard.

“In-Class” Exercises

We are running out of time, so if we don’t have time to discuss these in class or at the review, then I’ll distribute solutions.

1. Suppose language $L$ is recognizable. Argue $L \leq_p A_{TM}$.

2. Argue that 3SAT remains NP-complete, even if we require that each clause has exactly three literals, no constants, and no repeated variable.

3. The book defines NP as the class of languages with a polynomial time verifier $V$: for some polynomial $p(n)$, $V$ decides $\langle w, c \rangle$ in time at most $p(|w|)$. Suppose we change the definition, and bound $V$’s time by a polynomial in its total input length, $p(|\langle w, c \rangle|)$. Show that for any recognizable language $L$, there is such a verifier. (Hint: use histories.)

4. Suppose we know that 3COLOR is NP-complete (see below). Argue that 4COLOR is also NP-complete.

5. Problem 7.18: If P=NP, then almost every language in P is NP-complete. (Hint: map to one of two strings.)

Written Problems

Write up at least four of these, including at least two “reduction” problems (showing that something is NP-hard or NP-complete). The problem numbers here refer to the 3rd edition, they may differ in earlier editions. Unless the problem explicitly states otherwise, you should not assume earlier problem results to prove later problem results. You may assume any result argued in the text of Chapter 7.
Problem 1. Suppose we have access to a blackbox function $f$ solving SAT in linear time. That is, for a boolean formula $\phi$, $f(\langle \phi \rangle)$ returns 0 if $\phi$ is not satisfiable, and it returns 1 if it is. Using $f$ as a subroutine, show how we can find an actual satisfying assignment for $\phi$, when it is satisfiable, in polynomial time.

Problem 2. Problem 7.24, about CNF$_2$ and CNF$_3$.

Problem 3. Problem 7.25, showing CNF$_H$ is in P.

Problem 4. Problem 7.28, showing PUZZLE is NP-complete.

Problem 5. Problem 7.29, showing 3COLOR is NP-complete.

Problem 6. Problem 7.31, showing exam scheduling is NP-complete.

Problem 7. Problem 7.32, showing minesweeper consistency is NP-complete.

I may add a few more problems to this list.