Due September 12

Homework 1

Math 167 / CS 142: Complexity Theory

1. Problem 2.8.7(b) of the text.

> Suppose that we have a Turing machine with an infinite *two-dimensional string* (blackboard?). There are now moves of the form $\uparrow$ and $\downarrow$, along with $\leftarrow$ and $\rightarrow$. The input is written initially to the right of the initial cursor position.
>
> (b) Show that such a machine can be simulated by a 3-string Turing machine with a *quadratic* loss of efficiency.

(Here, "*quadratic* loss of efficiency" means that what the simulated machine does in the first s steps, the simulating machine accomplishes in its first $O(s^2)$ steps.)

---

2. Suppose that we have a Turing machine with a single string, but with two independently movable cursors on it. (Each cursor can read the symbol of the string at which it is positioned, whether that symbol was written by it or by the other cursor.) Show that such a machine can be simulated by a 2-string Turing machine (with each string having its own cursor, as usual) with a *linear* loss of efficiency. (Here, "*linear* loss of efficiency" means that what the simulated machine does it its first s steps, the simulating machine accomplishes in its first $O(s)$ steps.)

---

3. Problem 2.8.17 of the text.

> Show that any language decided by a k-string nondeterministic Turing machine within time $f(n)$ can be decided by a 2-string nondeterministic Turing machine *also* within time $f(n)$. (Discovering the simple solution is an excellent exercise for understanding nondeterminism. Compare with Theorem 2.1 and Problem 2.8.9 for deterministic machines.)
>
> Needless to say, any k-string nondeterministic Turing machine can be simulated by a single-string nondeterministic machine with a quadratic loss of efficiency, exactly as with deterministic machines.