1. P-complete? Say that we defined a problem to be "P-complete" analogously to our definition of NP-completeness (recall that definition!). Which problems in P are P-complete?

2. All Or Nothing. The problem AllOrNothingSat asks, given a 3-CNF Boolean formula (each clause has length exactly 3), whether there is an assignment to the variables such that each clause either has three TRUE literals or has three FALSE literals. Describe a polynomial-time algorithm to solve it.

3. Vertex Cover on Trees. Show that the VERTEX COVER problem is polynomially solvable on trees. What if we're interested in the weighted version of the problem – each vertex has a weight, and we want to find a vertex cover of minimum possible weight?

4. Self-reducibility – Decisions Problems Are Enough. We've started with this last time, but here are some fresh problems.
Show that it is enough to efficiently solve a decision problem to actually find a solution (in poly time) for the following problems. (One calls this "self-reduction" because you reduce the problem of finding a solution to the problem of deciding whether on exists.)

  (1) 3-COLORABILITY
  (2) HAMILTONIAN PATH
  (3) SUBSET SUM
  (4) $\mathbf{Ax = 1}$