CS302 - Assignment 21

Due: Thurday, May 10 at the beginning of class
Hand-in method: paper

"I can't find an efficient algorithm, but neither can all these famous people."

http://asfarian.wordpress.com/2011/01/11/np-completeness/

1. [**4 points**] All of the NP-Complete problems we looked at in class were *decision* problems, in that we were trying to decide if a condition was true or not (e.g. "does the graph have a clique of size k?"). However, for real problems we often would want to solve the max/min problem (often called the *search* problem), for example " what is the largest clique in the graph?".

   I claim that in most situations if you can solve the decision problem in polynomial time then you can solve the search problem in polynomial time. Prove that this is true for the CLIQUE problem.

2. [**12 points**] DOUBLE-SAT is the following problem: Given a boolean formula of n boolean variables $x_1, x_2, ..., x_n$ joined by m boolean connectives (one of: $\wedge$ (AND), $\vee$ (OR) and $\neg$ (NOT)), are there **two** different assignments of the variables such that the boolean formula evaluates to 1 (i.e. true)? For example, $(x_1 \vee \neg x_1 \vee \neg x_2) \wedge (x_2 \vee x_3) \wedge (\neg x_3)$ has two valid assignments, $x_1 = 1, x_2 = 1, x_3 = 0$ and $x_1 = 0, x_2 = 1, x_3 = 0$

   Prove that DOUBLE-SAT is NP-Complete. You may use any of the NP-Complete from the book or that we stated in class were NP-Complete.