Practice Final

1. [TRUE/FALSE/UNKNOWN] If VERTEX-COVER has a polynomial time algorithm then 3-SAT has a polynomial time algorithm.

2. [TRUE/FALSE/UNKNOWN] There is a polynomial time algorithm to solve the INDEPENDENT SET PROBLEM.

3. [TRUE/FALSE/UNKNOWN] Given a graph G with n vertices, deciding if there is a clique of size 195 is solvable in polynomial time.

4. [TRUE/FALSE/UNKNOWN] All problems in the class NP can be reduced to INDEPENDENT-SET.

5. Palindrome substring. Given a string, find the longest palindrome substring. Here is an example string. Let us identify all the palindromic substrings here.

   Input String : ACECCBA

   One letter palindrome substring A, B, C, E Two letter palindrome substring CC Three letter palindrome sequence CEC

   Give a Dynamic program to compute the length of the longest palindromic substring.

6. Profs for Dinner: You want to invite some Profs from the CSE Department for dinner (wonder why?!). As is well known, Profs are jealous of each other and end up not even on speaking terms. A grad student in the local sociology department created a database which is a two dimensional $n \times n$ matrix H (where n is the total number of CS Profs) with $H[i, j] = 1$ if Prof i is on speaking terms with Prof j and 0 otherwise (there are many 0s but at least lets assume H[i, i] = 1 for all i!) To make a good party, you want to invite at least k Profs, but so that all are willing to talk to each other.

   First, show that the problem of deciding if there is such a group of Profs is in the class NP.

   Second show that this problem NP-complete by reducing INDEPENDENT SET to this problem.

7. Suppose that someone gives you an oracle (a magic algorithm whose running time you can consider to be constant) to answer the CLIQUE decision problem: Does graph G have an clique of size k?. Show how to use this to solve for the actual elements of the largest clique in Graph G. If there are multiple different cliques of the same (largest) size, you can return any of them. Give the run-time of your algorithm (assuming the the oracle runs in constant time), and argue that your algorithm is correct.

8. For a graph G(V,E), an "ALMOST INDENDENT SET" (AVC) is a set of nodes V' where there is at most one edge connecting nodes in V'. Prove that "ALMOST INDEPENDENT SET" is NP-Complete.

   (a) Prove ALMOST INDEPENDENT SET is in the SET NP

   (b) Describe how to reduce a regular Vertex Cover problem to an ALMOST INDEPENDENT SET problem.

   (c) Argue that the solution to the regular Independent Set problem on the original input is always the same as the solution to the ALMOST INDEPENDENT problem you create.

   (d) Argue that your translation to create the ALMOST INDEPENDENT problem is polynomial time.