CS 577: Introduction to Algorithms	Homework 9
Out: 04/27/16	**Due: 05/04/16**

Ground Rules

- See HW1.
- All problems other than problem 0 will be graded.

Problems

0. Please fill out the online course evaluation form following the link you received in an email from AEFIS on Monday, Apr 25.

1. A subset S of vertices in an undirected graph G is called triangle-free if, for every triple of vertices $u, v, w \in S$, at least one of the three edges uv, uw, vw is absent from G. Prove that finding the size of the largest triangle-free subset of vertices in a given undirected graph is NP-hard.

   Hint: reduce Independent Set to this problem.

2. The *Not-All-Equal-SAT (NAE-SAT)* is a variant of 3-SAT that is also known to be NP-complete. The input to NAE-SAT is a 3-CNF formula. The formula is a Yes-instance if there exists an assignment to the variables such that every clause contains at least one true and at least one false literal. Otherwise the formula is a No-instance. For example, the formula $(x \vee y) \wedge (\bar{x} \vee \bar{y})$ is a Yes-instance because the assignment $x = T$ and $y = F$ satisfies it in the NAE manner. But the formula $(x \vee y) \wedge (\bar{x} \vee y)$ is a No-instance because for every assignment to x and y at least one of the two clauses are going to have both their literals true or both false.

   The *Max-Cut* problem is the opposite of (global) Min-Cut problem: the goal is to find a partition of the vertex set of an unweighted graph with the maximum number of edges going across. An instance of the Max-Cut problem is an unweighted graph G and a target integer k. The instance is a Yes-instance if it is possible to partition the vertex set of G into two parts such that at least k edges cross the cut; If no such partition exists, then the instance is a No-instance.

   Prove that Max-Cut is NP-Complete by giving a reduction from NAE-SAT to Max-Cut. You may assume that NAE-SAT is NP-Complete. (Don't forget to prove that Max-Cut is in NP.)

3. **(Extra Credit)** In this question we consider the special case of the Max-Cut problem with $k = |E|/2$. Surprisingly a $1/2$-Cut, that is, a cut with at least $|E|/2$ edges, always exists. Give a polynomial time algorithm for finding such a $1/2$-Cut.