1. Give a polynomial time algorithm that takes any tree T as input and computes a minimum vertex cover of T.

2. Give a graph G such that the cardinality of the vertex cover produced as output of our maximal matching based approximation algorithm (for the minimum vertex cover problem) on instance G is twice the cardinality of a minimum vertex cover of G.

3. For each of the following condition, give an infinite family of graphs $\mathcal{F}$ such that for each graph $G \in \mathcal{F}$,
   - vertex cover number of G is equal to the size of its maximum matching.
   - in addition to the above condition, G is also non-bipartite.
   - vertex cover number of G is twice the size of its maximum matching.

4. Consider the following factor 2 approximation algorithm for the cardinality vertex cover problem. Find a depth first search tree in the given graph, G, and output the set, say S, of all the non-leaf vertices of this tree. Show that S is indeed a vertex cover for G and $|S| \leq 2\ OPT$.
   `Hint:` Show that G has a matching in which all non-leaf vertices (and may be some leaf vertices) of the dfs tree are matched. What is the least size of such a matching?

5. Give a 1/2 factor approximation algorithm for the following : Given a directed graph $G = (V, E)$, pick a maximum cardinality set of edges from E so that the resulting subgraph is acyclic. Prove the approximation guarantee.
   `Hint:`Arbitrarily number the vertices. Consider the two sets of edges: (i) those from higher numbered vertices to lower numbered vertices and (ii) the remaining edges. What useful property of these sets is relevant for us? What scheme are you using for upper bounding OPT?

6. Consider the max-cut problem described as follows: Given a graph G (with at least two vertices) as input, find a partition of its vertex set into two non-empty subsets V_1 and V_2 such that the number of edges between the two parts is maximized. Number the vertices arbitrarily as $v_1, v_2, \ldots, v_n$. Insert v_1 to V_1 and v_2 to V_2. For each subsequent vertex, considered in the order of vertex numbers, insert it into V_1 or V_2, so that the number of edges from the new vertex to sets V_1 and V_2 that crosses the cut, is maximized. Show that this is a 1/2 factor approximation algorithm for the problem. What scheme are you using for upper bounding OPT?