HOMEWORK 11

Q1: Given an undirected graph G = (V, E), and positive integer k, the max-degree-spanning-tree problem asks whether G has a spanning tree whose degree is at most k. The degree of a spanning tree T is defined as the maximum number of neighbors a node has within the tree (i.e., a node may have many edges incident on it in G, but only some of them get included in T). Show that the max-degree-spanning-tree (MDST) problem is NP-complete. (20pts)

Ans:

Certificate: The certificate for Max Degree Spanning Tree is Spanning tree.

Verification: Suppose we have a spanning tree T of G, we can verify that the degree of T is at most k in polynomial time by checking the degree of each node in T. This can be done in O(V), where V is number of vertices in the graph.

Here, we can reduce vector cover problem to prove that MDST is NP complete.

Reduction:

For each edge in the original graph G, we create a new node v'. This helps us represent each edge as a distinct node in the new graph. For each vertex in original graph G, we create a new node u'. This helps us represent each vertex as a distinct node in the new graph.

Now, we connect nodes corresponding to edges and vertices. For each edge e = (u, v), we have corresponding node v' representing its endpoints u and v. Hence, we create edges between the node corresponding to one endpoint of edge e and the node corresponding to other endpoint of edge e. We also create a self-loop edge from v' to itself. Set k' = k - 1.

Claim:

G has a vertex cover of size at most k if and only if the constructed graph has a max degree spanning tree of degree at most k'.

Proof:

If G has a vertex cover of size at most k, then we can construct a max-degree-spanning-tree by selecting all the original vertices and their corresponding nodes in the constructed graph. Since each edge in G is incident to at least one selected node, and each selected node has degree at most k, the resulting tree has a maximum degree of at most k'.

If the constructed graph has a max-degree-spanning-tree of degree at most k', then the corresponding nodes in G form a vertex cover of size at most k. Since each edge in G corresponds to a node in the constructed graph, and each node in the spanning tree covers at least one incident edge, the selected nodes form a vertex cover.

This runs in polynomial time. Since Vertex cover problem can be reduced to this problem, therefore MDST problem is also NP Complete.

Q2: The k-cycle-decomposition problem for any k > 1 works as follows. The input consists of a connected graph G=(V, E) and k positive integers a1 , …, ak < |V|. The goal is to determine whether there exists k disjoint cycles of sizes a1 , …, ak respectively, s.t., each node in V is contained in exactly one cycle. Show that this problem is NP-complete (for any k > 1). (20 pts)

Ans:

Certificate: The certificate here is a set of k disjoint cycles, each of size a1 , …, ak covering all nodes in the graph exactly once.

Verification: Here, we have to check if each cycle has exactly ai vertices, all cycles are disjoint, and if each vertex is included in exactly one cycle. This can be done in polynomial time.

Here, we can reduce Hamilton Cycle problem to prove that this problem is NP complete.
Reduction:
For each edge in graph G, replace it with a path of length 2, adding two new vertices to it. This ensures that each edge in G corresponds to a cycle of size 2 in new graph G'. Set k' = 2.

Claim:
Given an instance of the Hamiltonian cycle problem, we can construct an equivalent instance of the k-cycle-decomposition problem.

Proof:
If G has a Hamiltonian cycle, then it means all vertices are connected in a cycle. This cycle can be represented as two disjoint cycles in G'.

If G' has 2 disjoint cycles covering all vertices, then removing one edge from each cycle yields a Hamiltonian cycle in G.

This runs in polynomial time. Since Hamilton cycle problem can be reduced to this problem, therefore this problem is also NP Complete.

Q3. Given a graph G = (V, E) with an even number of vertices as the input, the HALF-IS problem is to decide if G has an independent set of size |V| / 2. Prove that HALF-IS is in NP-Complete. (20pts)
Ans:
Certificate: The certificate for this problem is an independent set S of size |V|/2 in the graph G.

Verification: Here, we have to check if S contains exactly |V|/2 vertices, and no two vertices in S are adjacent. Both of these tasks can be done in polynomial time.

Here, we reduce Clique problem to prove that HALF-IS is NP Complete.
Reduction:
Here, if the original graph G has n vertices, we add (k-n) isolated vertices to G. This ensures that the modified graph G' has exactly k vertices. Set k' = k.

Claim:
Given an instance of the Clique problem, we can construct an equivalent instance of the HALF-IS problem.

Proof:
If G has a clique of size k, then G' has an independent set of size k (which is the complement of the clique in G'). Thus, G' is a "yes" instance of HALF-IS.

If G does not have a clique of size k, then G' does not have an independent set of size k. Thus, G' is a "no" instance of HALF-IS.

This can be done in polynomial time. Since Clique problem can be reduced to HALF-IS problem, therefore HALF-IS problem is also NP Complete.