26. (2 points) Consider the following three decision problems:

    DoubleFixedHamiltonianPath The input is an undirected graph G and two different vertices s and t in G. The problem is to determine if there is a simple path between s and t in G that spans all the vertices in G. A path is simple if it doesn't include any vertex more than once.

    SingleFixedHamiltonianPath The input is an undirected graph G and a vertex s in G. The problem is to determine if there is a simple path with one endpoint at s that spans all the vertices in G.

    HamiltonianCycle The input is an undirected graph G. The problem is to determine if there is a simple cycle that spans all the vertices in G.

    Show that if any one of these problems have a polynomial-time algorithm then they all have polynomial time algorithms.

27. (4 points) Consider the following three decision problems:

    DoubleFixedHamiltonianPath The input is an undirected graph G and two different vertices s and t in G. The problem is to determine if there is a simple path between s and t in G that spans all the vertices in G. A path is simple if it doesn't include any vertex more than once.

    SingleFixedHamiltonianPath The input is an undirected graph G and a vertex s in G. The problem is to determine if there is a simple path with one endpoint at s that spans all the vertices in G.

    HamiltonianCycle The input is an undirected graph G. The problem is to determine if there is a simple cycle that spans all the vertices in G.

    Show that if any one of these problems have a polynomial-time algorithm then they all have polynomial time algorithms using reductions that only make one call to the problem being reduced to.

28. (6 points) Show that if one of the following problems has a polynomial time algorithm then they all do.

    - The input is two undirected graphs G and H. The problem is to determine if the graphs are isomorphic.
    - The input is two undirected bipartite graphs G and H. The problem is to determine if the graphs are isomorphic. A graph is bipartite if the vertices can be colored with two colors so that no pair of adjacent vertices are colored the same color.
    - The input is two undirected Eulerian graphs G and H. The problem is to determine if the multigraphs are isomorphic. A graph is Eulerian if it has a closed walk that visits each edge exactly once (or equivalently the graph is connected and every vertex has even degree).
    - The input is two undirected regular graphs G and H. The problem is to determine if the graphs are isomorphic. A graph is regular if all vertices have the same degree, which is the number of adjacent vertices.
    - The input is two undirected Hamiltonian graphs G and H. The problem is to determine if the graphs are isomorphic. A graph is Hamiltonian if it has a Hamiltonian cycle, which is a simple cycle that spans the vertices.
    - The input is two undirected hypergraphs G and H. The problem is to determine if the hypergraphs are isomorphic. In a hypergraph the edges, called hyperedges, are collections of two or more (not just two) vertices.

    Intuitively, two graphs are isomorphic if on can name/label the vertices so that the graphs are identical. More formally, two undirected graphs G and H are isomorphic if there is a bijection f from the vertices of G to the vertices of H such that (v, w) is an edge in G if and only if $(f(v), f(w))$ is an edge in H. More formally, two undirected hypergraphs G and H are isomorphic if there is a bijection f from the vertices of G to the vertices of H such that $\{v_1, \ldots, v_k\}$ is a hyperedge in G if and only if $\{f(v_1), \ldots, f(v_k)\}$ is a hyperedge in H.

    NOTE: There is no known polynomial time algorithm for the graph isomorphism problem. For reasons that are too complicated to go into here (but I covered in CS 1511), it would be surprising if the graph isomorphism problem is NP-hard.