CIT 596 Online **Spring 2020**

Module 13–14

Name: NAME HERE

Questions requiring written answers.

1. We know that the subset sum problem is NP-Complete. Remember that the subset sum problem begins with an array of positive integers A and a targeted sum k. It then asks whether there exists a subset of A that sums up to this target value k. We would like to show that the following problem called Zero sum is also NP-Complete. Given a set of integers, is there a non-empty subset whose sum is zero. Show that the zero sum problem is in NP-Complete.

2. The PARTITION problem is as follows:

   Instance: A (multi-)set of numbers $S = \{a_1, a_2, \ldots, a_n\}$.

   Question: Can S be partitioned into two (multi-)sets A and B such that the sum of the numbers in A is equal to the sum of the numbers in B?

   Prove that PARTITION is NP-complete.

3. Exercise 5 of Chapter 8 on pages 506-507

4. We say that graph $G_1 = (V_1, E_1)$ is isomorphic to graph $G_2 = (V_2, E_2)$ if there is a bijective mapping f from V_1 to the vertices of G_2 such that V_2 such that $(u, v) \in E_1$ if and only if $(f(u), f(v)) \in E_2$. The SUBGRAPH ISOMORPHISM problem is as follows:

   Instance: Two graphs G and H.

   Question: Does H have a subgraph H' such that G is isomorphic to H'?

   Prove that SUBGRAPH ISOMORPHISM is NP-complete.

5. A Hamilton path in a graph is a simple path that visits all vertices and a Hamilton cycle is a simple cycle that visits all vertices. The HAMILTON PATH (respectively, HAMILTON CYCLE) problem is the following:

   Instance: An undirected graph G.

   Question: Does G contain a Hamilton path (respectively, Hamilton cycle)?

   (a) Assume without proof that HAMILTON PATH is NP-complete. Prove that HAMILTON CYCLE is NP-complete.

   (b) Now do the reverse: In other words, assume that HAMILTON CYCLE is NP-complete and prove that HAMILTON PATH is NP-complete. (This is the hard direction. It is easy to find a mapping that maps YES-instances of HAMILTON CYCLE to YES-instances of HAMILTON PATH. The challenge is to find a reduction that does this, but also maps NO-instances to NO-instances.

6. Suppose we are given a polynomial-time algorithm for solving the Hamilton Cycle decision problem. Describe how you could make just polynomially many calls to this algorithm to actually find a Hamilton cycle in a graph that has one.

7. The SET COVER problem is the following:

Instance: A set $U = \{1, 2, \ldots, n\}$ of n elements, a collection of subsets $S_1, S_2, \ldots, S_m$ of U, and an integer K.

Question: Are there K sets among the S_i's whose union is equal to U? In other words, are there K sets which together cover all the elements of U?

Starting with a problem that we have shown to be NP-complete, prove that SET COVER is NP-complete.