**LAB14**

**Jimmy Palma**

**610756**

1. **Show that TSP is NP-complete. (Hint: use the relationship between TSP and HamiltonianCycle discussed in the slides. You may assume that the HamiltonianCycle problem is NP-complete.)**

R (poli)→ HC (poli)→ TSP

HC is NP-COMPLETE, for transivty TSP IS np-problem

1. **True/False. Explain.** 
   1. **If Problem A is polynomial reducible to B and A is in NP, then B is in NPH.**

False, it could be NPH but not alwyas, if A is in NP and it is reducible to B, this means B is at least harder than A

* 1. **If Problem A is polynomial reducible to Problem B, then B is polynomial reducible to A.**

False, if A polinomial reducible to B, B is at least tahn A, means A is not harder, if B reduce to A, A would be harder

* 1. **If someone can find a polynomial time algorithm to solve one of the NP- Complete problems, then all NP-complete problems can be solved in polynomial time.**

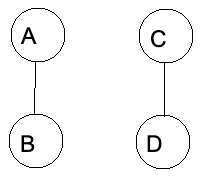
true

* 1. **Suppose A is an NP-complete problem and A is polynomial reducible to B. Then B is also NP-complete.**

true

1. **Show that the worst case for VertexCoverApprox can happen by giving an example of a graph G which has these properties:** 
   1. **G has a smallest vertex cover of size *s***
   2. **VertexCoverApprox outputs size 2\*s as its approximation to optimal size.**

Consider the following disconnected graph with 2 edges and 4 vertices:



The smallest vertez cover is A, C, but the algorithm return the four vertices

1. **The decision problem formulation of the Vertex Cover problem is this: Given a positive integer *k*, and a graph *G*, is there a vertex cover for *G* having size  *k*? Show that this decision problem belongs to *NP.***

For definition, the algorithm of vertex cover run in exponential O(2ⁿ) time, but verify the edge has a size of k runs is O(1). So it is hard to resolve but it is easy to verify

