# Prob1:

We already know: HamiltonianCycle → TSP & HamiltonianCycle problem is NP-complete

We only need to prove that TSP is NP-problem.

To verify that the TSP is NP, suppose we are given such a graph that does have a TSP cycle, and C is a solution. We determine the running time required to verify that C is indeed a TSP cycle. Here is what must be verified:

* Check: As a graph, C is a simple cycle.
* Check: C contains all vertices of G.
* Check: All edges of C are also edges of G.
* Check: Total weight of all edges in C <=k

These verification can be performed in O(n) steps. Therefore, TSP belongs to NP

So TSP is NP-complete!

# Prob2:

Show that the SubsetSum problem is polynomial reducible to this Knapsack problem.

Given a set **S** = {s0, s1, …, sn-1}, k (input for SubsetSum problem)

Obtain **KS =** {s0, s1, …, sn-1} (same with S) of n items, weights {w0, w1,…, wn-1}, values {v0, v1, …, vn-1}, a max weight W, and a min value V as bellow:

W = V = k

wi = vi = si

**Need to show**: S have a subset with sum k if and only if SS have a subset whose total value is no less than V and total weight is at most W

If T,k is a solution for S, T also a subset of KS, since wi = vi = si 🡪 sum(wi) = sum(vi) = sum(si) = k =W =V. That mean T is solution for KS.

If T is a solution for KS, then sum(wi)<=k and sum(vi) >=k. Since wi = vi = si 🡪 sum(si)=k. That mean T is also is solution for T.

# Prob3:

1. For this graph, size smallest Vertex cover is s=1
2. VertexCoverApprox output size is 2 = 2\*s

# Prob4:

To verify that the Vertex Cover is NP, we assume we are given input graph G(V,E) that has a solution, and that T is a solution. We determine the running time required to verify that T is a Vertex Cover of G and the size of T <= k.

Verify Algorithm

Vefity that size of T <= k //🡪 **O(1)**

For each e in E

(u,v) 🡨 getEndpoints(e)

belongTo(u,T) || belongTo(v,T) // 🡪 **O(m\*n)**

These verification can be performed in O(m\*n). Therefore, this problem belongs to NP