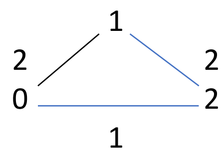
Ali Mirabzadeh
305179067
CS 180 – HW4
22.
Here is a counter example
Imagine the graph G below: It has two edges with the same weights for two edges, not distinct



The following set is not MST: {10, 12}: doesn't belong to MST which is a counter example
However, the set {10, 02} and {02, 12} are MST

Ali Mirabzadeh
305179067
CS 180 – HW4
25.
As the question suggests, we try to build a tree. I try to build a MST using the Kruskal's algorithm. On the way building the MST, we build τ inductively. Each time, I merge C_i and C_j by an edge of length l. I create the parent node, call it v, and the subtrees that has both C_i and $C_{j.}$ Also, we give it a height of h.
Note that for any P_i and $P_{j,}$ τ(pi , pj) is the same as the edge length considered when the components containing P_i and P_j were first joined. We have to show it is considered.
That would be proven by showing that since P_i and P_j will belong to the same component by the time the direct edge (pi , pj) is considered.
To show the second part, if τ' is any other hierarchical metric consistent with d, then τ'(pi , pj) ≤ τ(pi , pj) for each pair of points pi and pj.
We show this by contradiction.
Suppose that τ(pi , pj) ≤ τ'(pi , pj).
Let T' be a tree associated with τ', v' be the least common ancestor. T'_i and $T'_{j.}$would be the subtrees below v'. If h'is the height of v' in T', then (pi , pj) ≤ τ'(pi , pj) = h'_v
Consider the p_i-p_j in MST. We know that p_j is not in T'_I, there is a first node p' in P.
Let's p be the node that comes right after p' on P. Then d(p,p') is greater and equal h' which is greater since the least common ancestor of p and p' in T' would lie above the root node T'_i. bUT By the time Kruskal's algorithm merges the components containing pi and p_j all edges of P were Present; therefore, each has length at most τ(pi , pj) in which it contradicts the assumption.

Ali Mirabzadeh
305179067
CS 180 – HW4
28.
For part i, perform the regular kruskal's algorithm except that we should check if a visited node has labeled x or not, at the end if there was no MST, return false

```
Variable to count label with X
For each node in the graph
        Select edges in an increasing order from the beginning
        If edge is not labeled X
                Continue the loop
        Endif
        Check if the selected edge is creating a cycle or not
                minCost+=cost;
                do the union
                increment the count variable
        Endif
Endfor
If count variable is equal to K return true
Otherwise return false
```

Ali Mirabzadeh
305179067
CS 180 – HW4
30
To prove this, I claim that each extra node, Z, has degree of at least three in T; for if not, then the triangle inequality implies we can replace its two incident edges by an edge joining its two neighbors. Also, we know the sum of the degree in t-node tree is 2(t-1); in other words, every tree has at least as may leaves as it has nodes of degree greater than 2. If we get the MST on all sets of the union of X and Z where absolute of Z is less and equal to k, the cheapest among these will be minimum Steiner tree. There are at most n to the O(k); hence, the running time is n to O(k)