ALGORITHMS, FALL 2018, HOMEWORK 9

1. (a) This problem is identical to the problem solved by kruskal's algorithm. If we negate all the weights, we can just use Kruskal's algorithm to find the spanning tree with the maximal weights.

   The runtime is identical to Kruskal's algorithm, so:

   1. Make forest of vertices: $O(V)$ 2. Sort edges: $O(E \log E)$ 3. For each edge: $O(E)$ * $O(1)$ merge into mst.

   That makes the runtime at most $O(E \log E)$.

(b) The proof here will be done by contradiction.

Let us say that there exists a path with all of its edges greater than the smallest weight in T.

This cannot be possible, because we would choose the larger edge over the one that is smaller.

Therefore, there cannot exist a path with all of its edges greater than the smallest weight in T.