1. (5pts) How many edges does a tree with 14,381,132 vertices have? Do NOT need to explain your answer.

2. (15pts) Given the following tree

(a) What is the sequence of letters when the tree is traversed with preorder?

(b) What is the sequence of letters when the tree is traversed with inorder?
(c) What is the sequence of letters when the tree is traversed with postorder?
3. (15pts) Construct a complete binary (every node other than leaves have two children and all leaves are at the same level) tree whose traversal is a, b, c, d, e, f, g when

(a) it is traversed with preorder traversal

(b) it is traversed with inorder traversal

(c) it is traversed with postorder traversal
4. (20pts)

(a) Form a Binary Search Tree (BST) by inserting the numbers below in the order they appear in the sequence.

3, 7, 5, 1, 4, 9, 2

(b) What is the new BST after 7 is deleted from the above BST?
5. (10pts) Use Prim’s and Kruskal’s algorithms to find the MST in the following graph. Show the edge choices step by step as in Lecture note 9.
6. (20pts) Prove that:

The minimum spanning tree in a connected graph is unique if all weights are distinct.

(Hint: Assume there are more than one MST in the graph. Then, two distinct MSTs \( T_1, T_2 \) may share some edges, and the other edges either only exist in \( T_1 \) or only in \( T_2 \). Among those edges only in \( T_1 \) or only in \( T_2 \), we find the edge with smallest weight, denoted as \( e_x \). Without loss of generality, let’s say \( e_x \in T_1 \). Try add \( e_x \) to \( T_2 \) and see what happens.)

(Hint 2: simply saying the Prim’s or Kruskal’s algorithm will generate only one result does not prove this problem. Those algorithms generate ONE minimum spanning tree. They do not enumerate all possible minimum spanning trees of a given graph.)
7. (10pts) In the previous proof, what happens if not all weights are distinct? What part of the proof does not hold any more?