1. **What is Tree?**

A tree is similar to a linked list because it is a collection of nodes that point to one another. Unlike Linked list, trees point to in a non-linear fashion and can point to more than one node

1. **Why do we use Tree? Where can we sue tree data structure?**

We use trees because they organize our data in an order of rank in a non-linear way, because of this structure we are able to search for data faster, with binary search is O (logn). Trees can be used to index data (has fast retrieval). Trees can also be used to break down complex tasks into a smaller set of sub-tasks. Token based authentication w/ dependency.

1. **Can you list several types of tree data structure?**

There are Strict binary trees, Full binary tree and Complete binary trees.

1. **Strict Binary Tree**s- each Node has 2 children or no children
2. **Full Binary Tree** - each node has 2 children and leafs are at the same level
3. **Complete Binary tree**- Every level but last is filled, filled from left to right.
4. **What is Binary Tree?**

**Binary Tree**- A binary tree has 0, 1 or two children. Cannot have more than two children.

1. **What is binary search tree (BST)?**

**Binary Search Tree** is a node-based binary tree data structure

* The left subtree of a node contains only nodes with keys lesser than the node’s key.
* The right subtree of a node contains only nodes with keys greater than the node’s key.
* The left and right subtree each must also be a binary search tree.

1. **What are Leaves, Depth and Height?**

**Leave -** is a node with no children

**Depth:** The depth of the node is the distance from the root node to that particular node.(Bottom to top)(number of ancestors excluding itself)

**Height:**The height of the node is the distance from that node to the deepest node of that subtree.(Top to bottom)

1. **How many options are there for Tree Transversal?**

A tree can be transversed in **TWO** ways: Breadth first or Depth first.

1. **How do we transvers Breadth First Search (BFS) -Level Order?**

Breadth first - in a leveled order Starting at the root going down one level at a time. I n order to transvers breadth first we need to another data structure to hold the values while we visit each level. Usually QUEUE is used.

1. **How do we tranavers Depth First Search (DFS)?**

DFS starts at the depth and goes subtree by subtree based on the order you select.

1. **How do we transverse Pre-Order?**

Root, Left, Right

1. **How do we transverse In-Order?**

Left, Root, Right

1. **How do we transverse Post-Order?**

Left, Right, Root