Your Ultimate Guide To Landing Top AI roles

DECODE
AiML
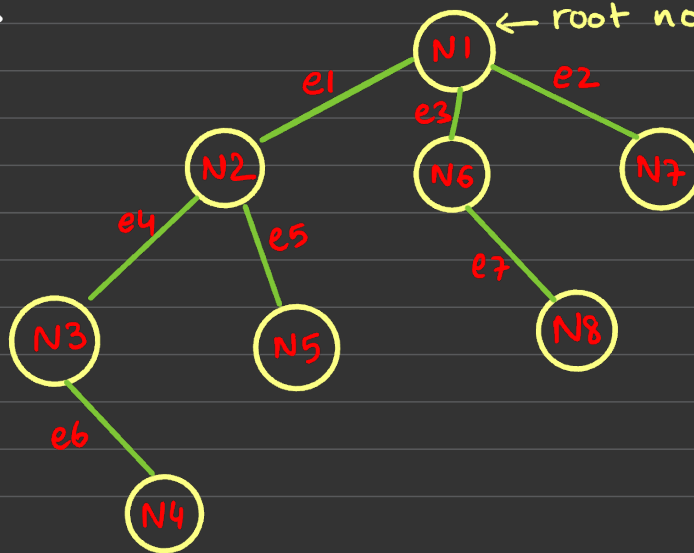
Introduction to Trees

→ A tree is a non-linear data structure that stores data in a hierarchical form.

→ It is non-linear (unlike array, stack, Queue, Linked Lists etc). so it allows branching.

→ It is made up of nodes connected by edges
  ① Nodes
  ② Edges

← root node

Nodes : N1, N2, N3, N4, N5, N6, N7, N8

Edges : e1, e2, e3, e4, e5, e6, e7

→ Real world examples of Trees
  ① File system in a Computer
  ② Recursion Tree
  ③ Organization Chart

<u>Basic Terminology</u>

→ Node :- A single element in a tree (Contains data + Reference to Children.

→ Root :- The topmost node (starting point)

→ Parent :- A node that has child nodes

→ Child :- A node derived from Parents

→ Edge :- The link/connection between two nodes.

→ Subtree :- A smaller tree inside a bigger Tree.

→ Degree :- The number of subtree of a node is called Degree.

→ Leaf Node ÷ A node with degree 0
    ↳ Also called Terminal Nodes

→ Degree of a Tree ÷ Maximum degree of nodes in the Tree.

→ Ancestor ÷ All nodes along the path from root to that node.

→ Descendent ÷ All nodes along the path from that node to leaf node.

→ Level number ÷ Root node assigned Level no 0 and every subsequent
    node is 1 more than Parent node.

→ Height/Depth of Tree ÷ Longest path from root to leaf.

→ Path ÷ Sequence of consecutive edges is called path.

Properties Of Trees

→ Tree is a graph without any cycles (Acyclic)

→ If there are N nodes, there will always be (N-1) edges.

→ A tree is always connected (all nodes reachable from root)

→ In computer science, we usually talk about rooted trees. Edges are implicitly directed from parent → child.

→ A tree with n nodes has at least 1 leaf node.

Types of Trees

① **General Tree**: Any node can have any no of children

② **Binary Tree**: Each node has at most 2 children (left, right)

③ **Binary Search Tree(BST)**: A binary tree with orderings.

Left child < Parent < Right child

④ **Balanced Trees**: AVL, Red-black Tree, B/B+ Tree

⑤ **Trie**: Tree for Strings/Prefixes.

⑥ **Heap**: A complete binary tree with some property (min/max heap)

⑦ **Fenwick Tree**: used in Prefix Sum Query, Range Sum Query.
   ↳ also called Binary Indexed Tree(BIT)

⑧ **Segment Tree**: used in Range Query (sum/min/max) Query.

Tree Implementation in Python

→ Tree Node

```
Class TreeNode (self, data)
        self.data = data
        self.children = []
```

→ Example usage

```
root = TreeNode ("Electronics")

laptop = TreeNode ("Laptop")
phone = TreeNode ("Phone")
  tv = TreeNode ("Tv")

root.children.append (laptop)
root.children.append (phone)
root.children.append (tv)
```
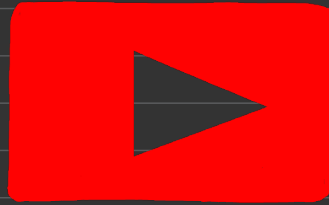
$\longrightarrow$ Upcoming Lectures

Binary Tree

Like Subscribe