Tree Traversals: Pre-order, In-order, and Post-order

Explanation and Answers for 10 Trees

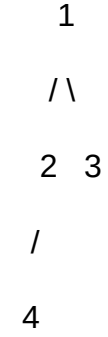
Tree 1:

- Pre-order: 1 2 3
- In-order: 2 1 3
- Post-order: 2 3 1

Explanation:

Pre-order visits the root first, followed by left and right. In-order visits the left, root, and then right. Post-order visits left, right, and finally the root.

Tree 2:



- Pre-order: 1 2 4 3
- In-order: 4 2 1 3

- Post-order: 4 2 3 1

Explanation:

Pre-order visits 1 (root), then moves to 2 (left subtree) and further down to 4. It then returns to visit 3 (right subtree).

In-order visits 4, then 2, then 1 (root), and finally 3. Post-order visits 4 first, then 2, then 3, and finally the root (1).

Tree 3:

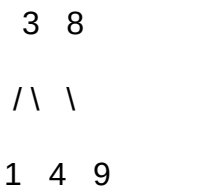
- Pre-order: 1 2 3 4 5
- In-order: 2 1 4 3 5
- Post-order: 2 4 5 3 1

Explanation:

In pre-order, we visit 1, then 2, then 3 and explore its left child 4 before its right child 5. In-order starts with 2, then root 1, then moves to 4, 3, and 5.

Post-order traverses the leftmost (2), then left subtree of 3 (4), then 5, before visiting 3 and finally 1.

Tree 4:



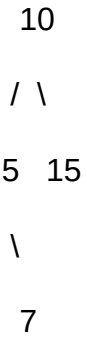
- Pre-order: 5 3 1 4 8 9
- In-order: 1 3 4 5 8 9
- Post-order: 1 4 3 9 8 5

Explanation:

Pre-order visits the root first (5), then left subtree (3 -> 1 -> 4), and then right subtree (8 -> 9). In-order traverses 1 -> 3 -> 4 -> 5 -> 8 -> 9.

Post-order explores the left subtree fully, then the right, and finally the root.

Tree 5:



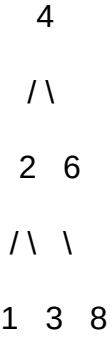
- Pre-order: 10 5 7 15
- In-order: 5 7 10 15
- Post-order: 7 5 15 10

Explanation:

Pre-order visits 10 (root), then moves to 5, goes to its right child 7, and finally explores 15. In-order traverses left subtree (5, then 7), then 10, and finally 15.

Post-order visits 7, then 5, then 15, and finally the root 10.

Tree 6:



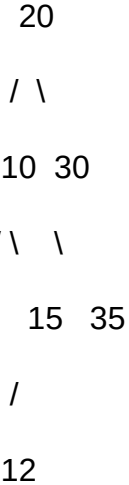
- Pre-order: 4 2 1 3 6 8
- In-order: 1 2 3 4 6 8
- Post-order: 1 3 2 8 6 4

Explanation:

Pre-order starts with root 4, then explores the left subtree (2 -> 1 -> 3), then the right subtree (6 -> 8). In-order traverses the left subtree, root, then right.

Post-order explores the left subtree, right subtree, and then root.

Tree 7:



- Pre-order: 20 10 5 15 12 30 35

- In-order: 5 10 12 15 20 30 35

- Post-order: 5 12 15 10 35 30 20

Explanation:

Pre-order visits the root first (20), explores the left subtree (10 -> 5 -> 15 -> 12), then the right subtree (30 -> 35). In-order traverses left, root, right.

Post-order explores leftmost, right subtree, then the root.

Tree 8:

- Pre-order: 1 2 4 5 3

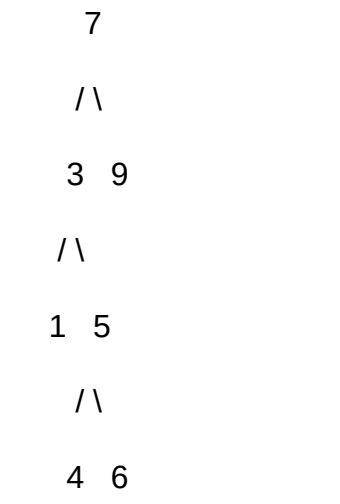
- In-order: 2 1 4 5 3

- Post-order: 5 4 2 3 1

Explanation:

Pre-order explores root 1, left subtree (2 -> 4 -> 5), then right subtree (3). In-order traverses left to right. Post-order goes from the bottom of the tree to the root.

Tree 9:

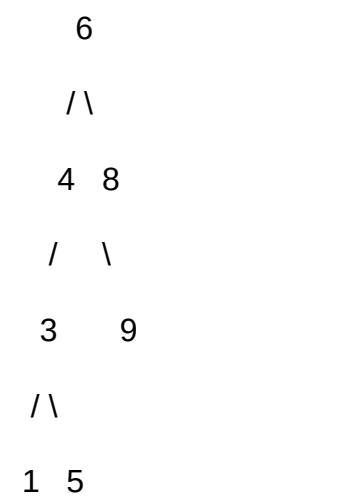


- Pre-order: 7 3 1 5 4 6 9
- In-order: 1 3 4 5 6 7 9
- Post-order: 1 4 6 5 3 9 7

Explanation:

Pre-order starts with 7 (root), explores left subtree (3 -> 1 -> 5 -> 4 -> 6), then right subtree (9). In-order traverses bottom left to right. Post-order explores fully then root.

Tree 10:



- Pre-order: 6 4 3 1 5 8 9
- In-order: 1 3 4 5 6 8 9

- Post-order: 1 5 3 4 9 8 6

Explanation:

Pre-order starts with root 6, explores left subtree (4 -> 3 -> 1 -> 5), then right subtree (8 -> 9). In-order goes from the left to right. Post-order explores leftmost, right subtree, and root.