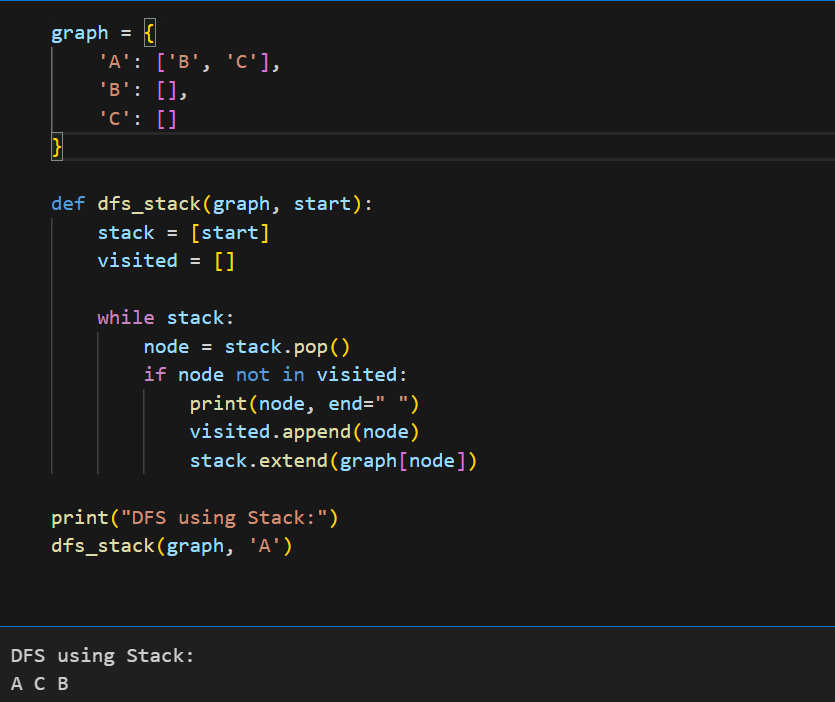
**MINAHIL QADIR**

**BSAI-3A(047)**

**TASK 5:**



This program performs Depth-First Search (DFS) using a stack.

A graph is represented as a dictionary where each node has a list of connected nodes.

dfs\_stack(graph, start) explores the graph starting from 'A'.

It uses a stack (LIFO approach) to keep track of nodes to visit.

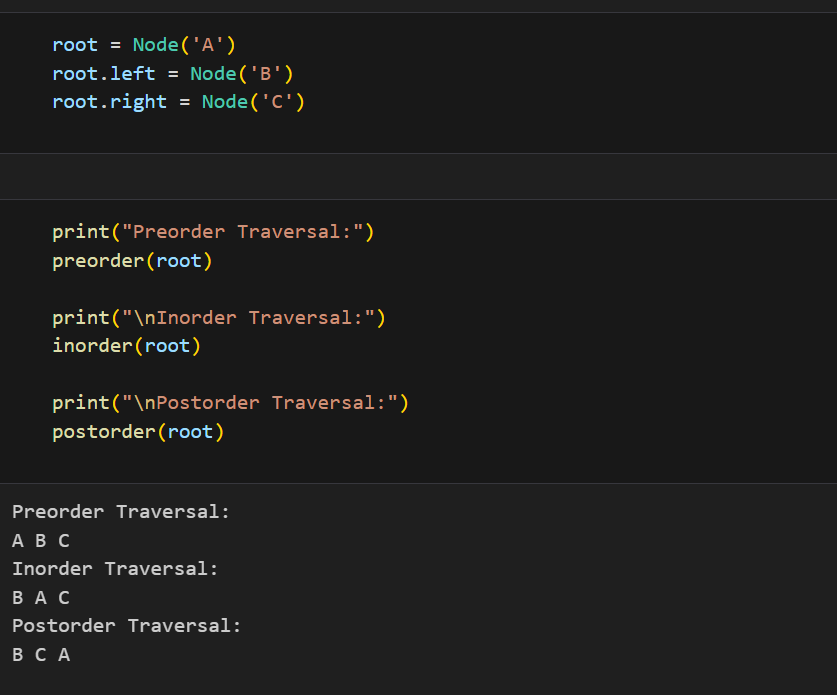
Steps of DFS traversal:

Pop a node from the stack.

If not visited, print it and mark it as visited.

Add its neighbors to the stack.

The function prints the DFS traversal order of the graph.

This program creates a simple tree and shows three ways to go through it.

Each node has a value and can have a left and right child.

The tree is built with A as the root, B as the left child, and C as the right child.

Three ways to visit the tree:

Preorder: Visit the root first, then left, then right.

Inorder: Visit left first, then root, then right.

Postorder: Visit left first, then right, then root.

The program prints the nodes in each order to show the difference.