AI LAB TASK 5

1. DFS with Stack & Node
2. 2. Research about "Inorder, Preorder, Postorder" and implement in DFS

Node Class:

Represents a binary tree node with:

`Value`: Stores the node's value.

`Left`: Reference to the left child (initially `None`).

`Right`: Reference to the right child (initially `None`).

Function: `dfs_with_stack(root)` performs Depth-First Search using a stack.

Check if root is `None`: If the tree is empty, return an empty list.

Initialize stack and result:

`Stack = [root]`: Start with the root node.

`Result = []`: List to store DFS traversal order.

While loop (process nodes):

Pop a node from the stack (LIFO).

Add the node's value to the `result` list.

Push right child (if exists) to the stack.

Push left child (if exists) to the stack.

Repeat until the stack is empty.

Return the DFS traversal order stored in `result`.


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
Python
PS C:\Users\Usman Ghani\Desktop\myworld> & "C:/Program Files/Python312/python.exe" "c:/Users/Usman Ghani/Desktop/myworld/AI_LAB_Task_5.py"
[1, 2, 4, 5, 3]
Inorder Traversal: [4, 2, 5, 1, 3]
Preorder Traversal: [1, 2, 4, 5, 3]
Postorder Traversal: [4, 5, 2, 3, 1]
PS C:\Users\Usman Ghani\Desktop\myworld>