**Name : Nabeeha Ali Rana**

**Section : DS-3A**

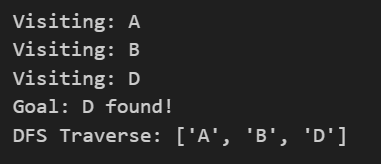
**Roll no : BSDSM - 037**

**AI Lab**

**Lab Task 5**

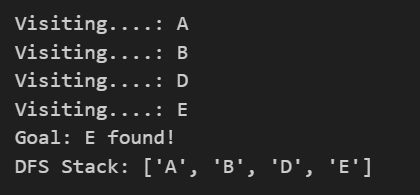
## **1. DFS with Dictionary (Tree Structure)**

* **How it works**:  
  + The tree is represented as a **dictionary** where each node maps to its children.
  + A **stack** is used to keep track of nodes to visit.
  + The algorithm pops the last node, visits it, and pushes its children (in reverse order).
  + It stops when the **goal node** is found or when the stack is empty.
* **Why it works**:  
  + DFS is naturally implemented with a stack because it explores one branch deeply before backtracking.
  + Reversing the children ensures nodes are visited in left-to-right order.
* **Output:**

****

## **2. DFS with Node Class**

* **How it works:**
  + Each node is represented as a Node object with a value and list of children.
  + The DFS algorithm works in the same way as the dictionary version, but instead of strings, it uses node objects.
  + When a node matches the goal value, the search stops and the visited path is returned.
* **Why it works:**
  + The class-based approach is object-oriented and more flexible for building and expanding trees.
  + It allows tree structures to be created dynamically rather than hardcoding them in a dictionary.
* **Output:**

****