**Artificial Intelligent (Lab)**

**Task # 05**

**A purple circle with dots and lines

AI-generated content may be incorrect.**

**Submitted To: Sir Rasikh Ali**

**Submitted By: Shumaila Maryam**

**Roll no: SU92-BSDSM-F24-062**

**Submitted On: 25-September-25**

**Section: BSDS-3A**

**Department of Software Engineering, Superior University, Lahore**

**Question # 01:**

**DFS with Stack**

**Introduction:**  
This Python program implements the Depth-First Search (DFS) algorithm to find a goal in a tree structure. Using a stack, it explores as far down a branch as possible before backtracking. The program prints each node as it visits it, clearly showing the path it takes until the goal is found.

**Why I Made This:**  
The goal of this project was to:

* Learn how the Depth-First Search algorithm works in practice.
* Practice using a stack to control the traversal of a tree.
* Understand how to represent and navigate a tree structure using a dictionary.
* Create a program that clearly visualizes each step of the pathfinding process.

**How It Works:**

**1. Tree Structure**  
The program starts with a tree defined as a dictionary, where each key is a node and its value is a list of its children:

**2. Initialization**  
The program keeps two lists:

* visit\_stack: To store the nodes that have already been visited.
* stack: To keep track of the nodes to visit next, starting with the root node 'A'.

**3. Traversing the Tree**  
The program uses a loop that continues as long as the stack is not empty:

* It pops the last node added from the stack.
* If the node has not been visited, it is added to the visit\_stack.
* It then checks if this node is the goal. If it is, the search stops.
* If it's not the goal, it adds all of its unvisited children to the stack, ensuring the search goes deeper into the tree.

**Summary:**  
This program shows a simple way to implement the Depth-First Search algorithm. Using a stack makes the program's logic clear and easy to follow. It’s useful for understanding how tree traversal works and provides a basic foundation for more complex pathfinding and graph-related problems.

**Output:**

A black screen with white text

AI-generated content may be incorrect.