Faculty of Computing and Information Technology

University of the Punjab,

Lahore

Artificial Intelligence Lab 01

Instructor: Qamar U Zaman

Search Algorithms - DFS, BFS, Iterative Deepening, and Beyond

Objective:

The objective of this lab is to implement and compare various search algorithms in artificial intelligence, focusing on DFS, BFS, and IDDFS. Additionally, the lab aims to explore practical considerations such as heuristic functions, informed search strategies, and their applications.

Tasks:

1. **Implementing Basic Search Algorithms:**
    - **DFS (Depth-First Search):**
        - Implement DFS to traverse a graph or search tree recursively.
        - Track visited nodes to prevent revisiting and handle cycles in graphs.
    - **BFS (Breadth-First Search):**
        - Implement BFS to explore all nodes at the present "depth" level before moving on to nodes at the next level.
        - Utilize a queue data structure to manage nodes to be visited.
    - **IDDFS (Iterative Deepening Depth-First Search):**
        - Implement IDDFS to combine benefits of BFS and DFS for optimal memory usage and completeness.
        - Perform DFS repeatedly with increasing depth limits until the goal is found.

2. **Enhanced Search Strategies:**
    - *Informed Search (A Algorithm):**
        - Implement A* search algorithm using a heuristic function to guide the search towards the goal efficiently.
        - Compare the performance of A* with the uninformed search algorithms (DFS, BFS, IDDFS).
    - **Greedy Best-First Search:**
        - Implement Greedy Best-First Search using a heuristic function that selects the node closest to the goal based on a heuristic estimate.
        - Discuss the limitations and advantages compared to A* and other algorithms.

3. **Applications and Analysis:**
    - **Pathfinding in a Maze:**
        - Apply each algorithm to find the shortest path through a maze represented as a grid.
        - Compare path lengths, computational effort, and feasibility for larger maze sizes.

    - **8-Puzzle Problem:**
        - Solve the 8-Puzzle using DFS, BFS, IDDFS, A*, and Greedy Best-First Search. ■ Analyze the number of moves, computational resources used, and effectiveness of each algorithm.