# A\* Algorithm

## How It Works

1. The graph is represented using an adjacency list where each node connects to other nodes with weighted edges.  
2. The heuristic function (h) estimates the remaining cost from each node to the goal.  
3. The algorithm maintains two sets: open\_list (nodes to visit) and closed\_list (visited nodes).  
4. For each node, it calculates f(n) = g(n) + h(n), selecting the node with the smallest f(n) value.  
5. Once the goal node is reached, the path is reconstructed from the parent nodes.

## Why We Use A\* Algorithm

• It combines Dijkstra’s Algorithm (for accurate path cost) and Greedy Best-First Search (for speed).  
• Guarantees the shortest path if the heuristic function is admissible.  
• Widely used in artificial intelligence, robotics, and GPS navigation systems.

## Output Screenshot

Path found: ['A', 'B', 'D']