# Lab 7: A\* Algorithm in Python

The A\* (A-star) Algorithm is a popular pathfinding and graph traversal method used in Artificial Intelligence. It finds the shortest path between a start node and a goal node by combining the cost to reach the node and the estimated cost to reach the goal (heuristic).

A\* works by maintaining two main lists:  
1. The Open List — which contains nodes to be evaluated.  
2. The Closed List — which contains nodes already evaluated.

## Working Steps:

1. Initialize the open list and closed list.  
2. Add the start node to the open list.  
3. Find the node with the lowest f(n) = g(n) + h(n), where:  
 • g(n) = actual cost from start to current node  
 • h(n) = estimated cost from current node to goal  
4. Move this node to the closed list.  
5. For each neighbor of this node, calculate cost and update path.  
6. Repeat until the goal node is reached.  
7. Reconstruct the path from goal to start using parent nodes.

## Heuristic Function (h):

The heuristic function estimates how close a node is to the goal. A good heuristic helps A\* find the optimal path efficiently. For example, in a grid, the heuristic can be the straight-line distance (Euclidean distance) between two points.

## Output Example:

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

## Conclusion:

The A\* algorithm is efficient and guarantees the shortest path when the heuristic is admissible (does not overestimate the cost). It is widely used in game development, robotics, and GPS navigation systems.