**Name: Muzamil Mushtaq**

**Roll No: SU92-BSAIM-F24-037**

**Section: BSAI-3A**

**Submitted to: Sir Rasikh Ali**

**Topic: A\* Algorithm**

**Introduction**

The A\* (A-Star) algorithm is one of the most popular and efficient pathfinding and graph traversal algorithms.  
It is widely used to find the shortest path between two nodes in a weighted graph, especially in maps, games, robotics, and AI systems.

A\* combines the strengths of Dijkstra’s Algorithm and Greedy Best-First Search:

* Like Dijkstra, it considers the actual distance traveled so far.
* Like Best-First Search, it uses a heuristic function to estimate the cost from the current node to the goal.

The algorithm minimizes the evaluation function:

f(n)=g(n)+h(n)f(n) = g(n) + h(n)f(n)=g(n)+h(n)

Where:

* f(n) estimated total cost of the cheapest solution through node *n*
* g(n) actual cost from the start node to *n*
* h(n) estimated cost (heuristic) from *n* to the goal node

**2. Working / Usage**

How it works

1. Start from a start node.
2. Keep a set of open nodes (to be evaluated) and closed nodes (already evaluated).
3. At each step, pick the node with the lowest f(n) = g(n) + h(n).
4. For each neighbor of the current node:
   * Calculate its new cost g(n).
   * If this new path is better, update its parent and cost.
5. Repeat until the goal node is reached or all nodes are explored.

**3. Advantages**

**Optimal and Efficient:**  
Finds the shortest path efficiently when the heuristic is admissible (never overestimates cost).

**Flexible:**  
The heuristic function can be customized based on the problem (e.g., Euclidean distance, Manhattan distance, etc.).

**Complete:**  
If there is a path, A\* will always find it (given an admissible heuristic).

**Practical:**  
Used in GPS navigation, video game AI (enemy movement), network routing, and robotics path planning.

**4. Disadvantages**

**Memory Intensive:**  
A\* maintains large open and closed lists making it unsuitable for very large graphs.

**Heuristic Dependency:**  
The quality of results depends heavily on how accurate or appropriate the heuristic function is.

**Performance Drop in Complex Maps:**  
If the heuristic is weak or misleading, A\* can become almost as slow as Dijkstra’s algorithm.