**Questions/Answer:**

**1. Which algorithm gives the shortest path? Why?**

* **BFS and A\*** both found the shortest path (3 steps).
* **DFS did not**, because DFS explores depth-first without checking path cost.
* BFS guarantees shortest path in an unweighted graph since it explores level by level.
* A\* guarantees shortest path too, because the heuristic is admissible (never overestimates distance to goal).

**2. Which algorithm is more efficient in terms of time and memory?**

* **BFS**:
  + Time: explores states level by level, may expand many unnecessary states.
  + Memory: can be large, since it stores entire frontier of states.
* **DFS**:
  + Time: may take longer if unlucky (goes deep into useless paths).
  + Memory: more efficient (stack only holds current path).
  + But not guaranteed to find shortest path.
* **A\***:
  + Time: generally faster than BFS if the heuristic is good.
  + Memory: similar to BFS (keeps open list), but guided search reduces expansions.
  + Best balance between optimality and efficiency.

**📌 Final Summary**

* **Shortest Path** → BFS & A\* (3 steps).
* **Most memory efficient** → DFS.
* **Best trade-off (optimal + efficient)** → A\*.

Top of Form

Bottom of Form