

8 puzzle Problem

Given a 3×3 board with 8 tiles (each numbered from 1 to 8) and one empty space, the objective is to place the numbers to match the final configuration using the empty space. We can slide four adjacent tiles (left, right, above, and below) into the empty space.

1. 8 puzzle Problem using DFS (Brute-Force)

We can perform a depth-first search on state-space (Set of all configurations of a given problem i.e. all states that can be reached from the initial state) tree.

- Depth-first search on state-space tree.
- Successive moves may take us away from the goal.
- Inefficient as it explores all paths equally.

Idea: DFS explores as far as possible along each branch before backtracking. It is an exhaustive search technique used to traverse or search tree or graph data structures. It starts at the root node and explores as far as possible along each branch before backtracking.

Approach:

1. Start from the root node.
2. Explore the leftmost child node recursively until you reach a leaf node or a goal state.
3. If a goal state is reached, return the solution.
4. If a leaf node is reached without finding a solution, backtrack to explore other branches.

