1. 8 puzzle game
2. from collections import deque
3. def bfs(start, goal):
4. visited, queue = set(), deque([(start, [])])
5. while queue:
6. state, path = queue.popleft()
7. if state == goal: return path + [state]
8. visited.add(state)
9. i = state.index(0)
10. for move in [-3, 3, -1, 1]:
11. ni = i + move
12. if 0 <= ni < 9 and (i % 3 != 0 or move != -1) and (i % 3 != 2 or move != 1):
13. new = list(state)
14. new[i], new[ni] = new[ni], new[i]
15. tnew = tuple(new)
16. if tnew not in visited:
17. queue.append((tnew, path + [state]))
18. return None
19. start = (1, 2, 3, 4, 5, 6, 0, 7, 8)
20. goal = tuple(map(int, input("Enter goal state (9 numbers, space-separated): ").split()))
21. path = bfs(start, goal)
22. if path:
23. for step, state in enumerate(path):
24. print(f"\nStep {step}:")
25. for i in range(0, 9, 3): print(state[i:i+3])
26. else:
27. print("No solution found.")

OUTPUT:

