1. Which of the following is not a backtracking algorithm?
2. Knight tour problem
3. N queen problem
4. Tower of Hanoi
5. M coloring problem
6. Backtracking algorithm is implemented by constructing a tree of choices called:
7. State-space tree
8. State-chart tree
9. Node tree
10. Backtracking tree
11. In what manner is a state-space tree for a backtracking algorithm constructed?
12. Depth-first search
13. Breadth-first search
14. Twice around the tree
15. Nearest neighbor first
16. The leaves in a state-space tree represent only complete solutions.
17. True
18. False
19. The problem of finding a path in a graph that visits every vertex exactly once is called:
20. Hamiltonian path problem
21. Hamiltonian cycle problem
22. Sum of subset problem
23. Turnpike reconstruction problem
24. What happens when the backtracking algorithm reaches a complete solution:
25. It backtracks to the root
26. It continues searching for other possible solutions.
27. It traverses from a different route
28. Recursively traverses through the same route
29. Backtracking algorithm is faster than the brute force technique?
30. True
31. False
32. Complexity of generating all possible permutations using Backtracking is:
33. O(N!)
34. O(NxN)
35. O(N\*N!)
36. O(N)
37. Complexity of N-queens problem:
38. O(N^2N)
39. O(NxN)
40. O(N\*N!)
41. O(N^2)
42. What is the difference between backtracking and brute force?

A. Backtracking is a more efficient search algorithm than brute force.

B. Backtracking uses heuristics to guide the search, while brute force does not.

C. Backtracking is a more general approach that can be used to solve a wider range of problems than brute force.

D. Backtracking is a more constrained approach that only considers solutions that satisfy certain constraints, while brute force considers all possible solutions.

1. What are some common pitfalls to avoid when designing and implementing a backtracking algorithm?

A. Not defining the constraints clearly and precisely.

B. Failing to backtrack properly when a dead end is reached.

C. Not pruning the search space to eliminate unpromising solutions.

D. All of the above.