BACKTRACKING

1. [Backtracking | Set 1 (The Knight's tour problem)](#)
2. [Backtracking | Set 2 (Rat in a Maze)](#)
3. [Backtracking | Set 3 (N Queen Problem)](#)
4. [N Queen in O(n) space](#)
5. [Backtracking | Set 4 (Subset Sum)](#)
6. [Backtracking | Set 5 (m Coloring Problem)](#)
7. [Backtracking | Set 6 (Hamiltonian Cycle)](#)
8. [Backtracking | Set 7 (Sudoku)](#)
9. [Boggle | Set 2 (Using Trie)](#) (MICROSOFT)
10. [Remove Invalid Parentheses](#)
11. [Rat in a Maze with multiple steps or jump allowed](#)
12. [Write a program to print all permutations of a given string](#) (iski ekk or achi approach crack the coding interview book m diya h… dekh lena)
13. [Print all possible paths from top left to bottom right of a mXn matrix](#)
14. https://www.geeksforgeeks.org/find-number-of-islands/ (COMPANY)
15. [Tug of War](#)
16. [Combinational Sum](#)
17. [Backtracking to find all subsets](#)
18. [Find all distinct subsets of a given set](#)
19. [Power Set in Lexicographic order](#)
20. [Word Break Problem using Backtracking](#)
21. [Count all possible paths between two vertices](#)
22. [Partition of a set into K subsets with equal sum](#)
23. [Longest Possible Route in a Matrix with Hurdles](#)
24. [Match a pattern and String without using regular expressions](#)
25. [Fill two instances of all numbers from 1 to n in a specific way](#)
26. [Find shortest safe route in a path with landmines](#)
27. [Find paths from corner cell to middle cell in maze](#)
28. [Find if there is a path of more than k length from a source](#)
29. [Find Maximum number possible by doing at-most K swaps](#)
30. [Print all palindromic partitions of a string](#)
31. [Print all paths from a given source to a destination](#)
32. [Print all possible strings that can be made by placing spaces| Set-1](#)
33. [Print all possible strings that can be made by placing spaces| Set-2](#)
34. [Print all longest common sub-sequences in lexicographical order](#)
35. [Smallest expression to represent a number using single digit](#)
36. [Given an array A[] and a number x, check for pair in A[] with sum as x](#)