

Backtracking:

- 1) Dado um vetor de inteiros distintos, desenvolve uma função que retorna todas as permutações desse vetor. A ordem de como são retornadas não interessa. **

Ex :

```
1) Input: [1,2,3]
Output: [[1,2,3],[1,3,2],[2,1,3],[2,3,1],[3,1,2],[3,2,1]]
2) Input: [0,1]
Output: [[0,1],[1,0]]
3) Input: [1]
Output: [[1]]
```

- 2) Dado um certo número n de parênteses, desenvolve uma função que gera todas as combinações de parênteses bem formados. A ordem de como são retornados não interessa. **

Ex :

```
1) Input: n = 3
Output: ["((()))","(()())","(())()","()()()","()()()"]
2) Input: n = 1
Output: ["()"]
```

- 3) Dada uma string S, é possível transformar cada letra individualmente em uppercase ou lowercase para criar uma nova string. Desenvolve uma função que retorna uma lista de todas as strings possíveis de criar deste modo. A ordem de como são retornados não interessa. **

Ex :

```
1) Input: S = "a1b2"
Output: ["a1b2","a1B2","A1b2","A1B2"]
2) Input: S = "3z4"
Output: ["3z4","3Z4"]
3) Input: S = "12345"
Output: ["12345"]
4) Input: S = "0"
Output: ["0"]
```

- 4) O puzzle das n-rainhas é um problema com o objetivo de pôr n rainhas num tabuleiro de xadrez n x n tal que nenhuma rainha ataca outra e vice-versa. Relembra que uma rainha pode andar em todas as direções. Dado um inteiro n, retorna todas as soluções distintas para este puzzle. Cada solução contém uma configuração das n rainhas no tabuleiro diferente tal que "Q" indica rainha e "." indica um espaço vazio. ***

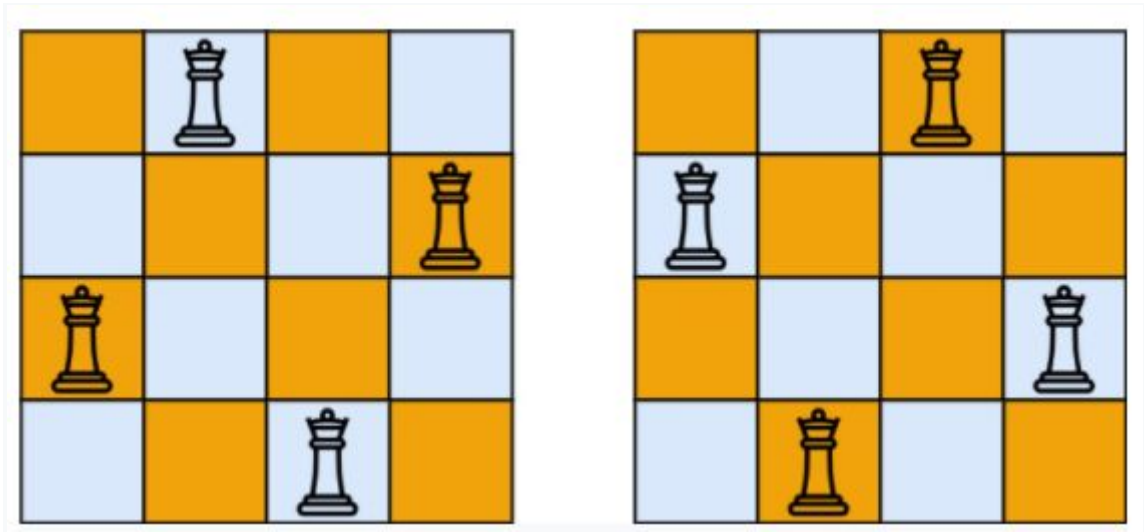
Ex:

1) Input: $n = 4$

Output: `[[".Q..","...Q","Q...","..Q."],["..Q.","Q...", "...Q",".Q.."]]`

Explicação: Existem duas soluções distintas para o puzzle de 4

rainhas. Cada sublista representa uma fila no tabuleiro.



2) Input: $n = 1$

Output: `[["Q"]]`