

Output:

```
Enter string to generate its permutations: ABC
ABC ACB BAC BCA CBA CAB
```

Interview Problem : N-Queen

What is the N-Queens Problem?

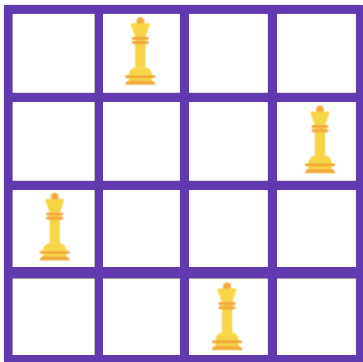
How can N queens be placed on an NxN chessboard so that no two of them attack each other?

Let's take an example of N=4.

Movement of Queen in a chessboard: A Queen can move in any direction of board but while moving, it can't change the direction.

Therefore two queens shouldn't be in same row and column so that they can't attack each other.

When N=4, the solution looks like :



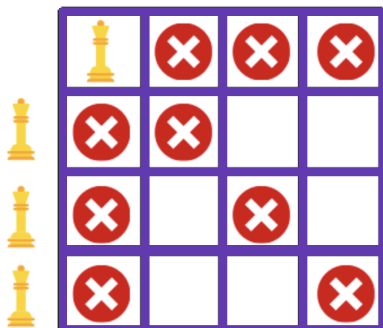
Solution to the N-Queen Problem

We're trying to fix this by putting the queen in place and removing the possibility of an attack on her. We place one queen in each row/column.

If you see that the queen is under attack at the selected location, try the next location.

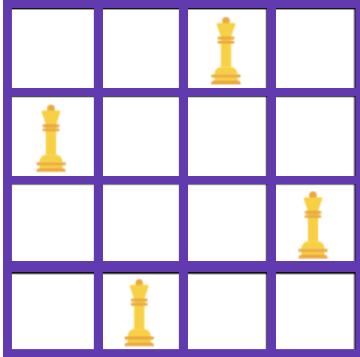
If the queen is under attack at every position in a row, go back and reposition the queen that was in front of the current position.

Repeat this process of placing and returning queens until all N queens have been successfully placed.



The red cross marks the positions which are under attack from a queen. Whenever we reach a state where we have a queen to place but all the positions in the rows are under attack, we backtrack.

This is not the only possible solution to the problem. If you move each queen one step forward in a clockwise manner, we get another solution.:



Code : [LP_CODE2.java](#)

Output.

```
0 0 1 0
1 0 0 0
0 0 0 1
0 1 0 0
```

Interview problem: Sudoku Solver

What is Sudoku

Simply put, Sudoku is a combinatorial number placement puzzle with a 9 x 9 grid of cells partially filled with numbers from 1 to 9. The goal is to fill the remaining empty fields with the remaining numbers so that there is only one. Numbers in each row and column. Quantity by type.

Also, each subsection of the 3 x 3 grid cannot have duplicate numbers.
Difficulty increases naturally with the number of empty fields on each board.

For example this is the unsolved Sudoku puzzle

```
8.....
..36.....
.7..9.2..
.5...7...
....457..
...1...3.
..1....68
..85...1.
.9....4..
```