N Queens Problem and its Java Implementation

Problem Statement:

The N Queens problem is a classical backtracking problem. The task is to place N queens on an N x N chessboard such that no two queens attack each other.

Algorithm:

1. Initialize a chessboard of size N x N with all cells set to 0.
2. Place a queen on the first row and first column of the chessboard.
3. For each row from 2 to N:
   a. For each column from 1 to N:
      i. If the cell is safe (i.e., no other queen is attacking it), place a queen on the cell.
      ii. If the cell is not safe, skip it.
4. Print the chessboard.

Java Implementation:

```java
import java.util.Arrays;

public class NQueens {

    private int N;
    private int[][] chessboard;

    public NQueens(int N) {
        this.N = N;
```

```
    chessboard = new int[N][N];
}

public boolean solve() {
    if (!solve(0)) {
        System.out.println("No solution found");
        return false;
    }
    return true;
}

private boolean solve(int row) {
    if (row == N) {
        return true;
    }

    for (int col = 0; col < N; col++) {
        if (isSafe(row, col)) {
            chessboard[row][col] = 1;

            if (solve(row + 1)) {
                return true;
            }

            chessboard[row][col] = 0; // Backtrack
        }
```

```
    }

    return false;
}

private boolean isSafe(int row, int col) {
    // Check row
    for (int i = 0; i < row; i++) {
        if (chessboard[i][col] == 1) {
            return false;
        }
    }

    // Check diagonal (left)
    for (int i = row, j = col; i >= 0 && j >= 0; i--, j--) {
        if (chessboard[i][j] == 1) {
            return false;
        }
    }

    // Check diagonal (right)
    for (int i = row, j = col; i >= 0 && j < N; i--, j++) {
        if (chessboard[i][j] == 1) {
            return false;
        }
    }
```

```
        return true;
    }

    public void printSolution() {
        for (int i = 0; i < N; i++) {
            System.out.println(Arrays.toString(chessboard[i]));
        }
    }

    public static void main(String[] args) {
        NQueens nQueens = new NQueens(4);
        if (nQueens.solve()) {
            nQueens.printSolution();
        }
    }
}
```

Example:

For a 4x4 chessboard, the solution is:

```
[0, 1, 0, 0]
[0, 0, 0, 1]
[1, 0, 0, 0]
[0, 0, 1, 0]
```