

AI Assignment 4 – Implementation of CSP Problem

Name : Siddhi Rajeshirke

Division : CS-C

Batch : 3

Roll No : 75

PRN : 12110298

Code :

```
import java.util.Scanner;
public class CSP_NQueens {
    int boardSize;

    CSP_NQueens(int size) {
        this.boardSize = size;
    }

    void displayChessboard(int[][] chessboard) {
        for (int row = 0; row < boardSize; row++) {
            for (int col = 0; col < boardSize; col++) {
                if (chessboard[row][col] == 1)
                    System.out.print("Q ");
                else
                    System.out.print(". ");
            }
            System.out.println();
        }
        System.out.println();
    }

    boolean isSafeToPlaceQueen(int[][] chessboard, int row, int col) {
        int i, j;

        for (i = 0; i < col; i++)
            if (chessboard[row][i] == 1)
                return false;

        for (i = row, j = col; i >= 0 && j >= 0; i--, j--)
            if (chessboard[i][j] == 1)
                return false;

        for (i = row, j = col; j >= 0 && i < boardSize; i++, j--)
            if (chessboard[i][j] == 1)
                return false;

        return true;
    }

    boolean solveNQueens(int[][] chessboard, int col) {
        if (col >= boardSize) {
            displayChessboard(chessboard);
            return true;
        }

        boolean foundSolution = false;
        for (int row = 0; row < boardSize; row++) {
```

```

        if (isSafeToPlaceQueen(chessboard, row, col)) {
            chessboard[row][col] = 1;
            displayChessboard(chessboard);

            foundSolution = solveNQueens(chessboard, col + 1) ||
foundSolution;
            chessboard[row][col] = 0;
        }
    }
    return foundSolution;
}

void findSolutions() {
    int[][] chessboard = new int[boardSize][boardSize];

    if (!solveNQueens(chessboard, 0)) {
        System.out.println("No solution exists");
    }
}

public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    System.out.print("Number of queens: ");
    int n = scanner.nextInt();
    CSP_NQueens solver = new CSP_NQueens(n);
    solver.findSolutions();
}
}

```

Output :

```
CSP_NQueens x
C:\Users\siddh\jdk\openjdk-19.0.1\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Com
Number of queens: 4
Q . . .
. . . .
. . . .
. . . .

Q . . .
. . . .
. Q . .
. . . .

Q . . .
. . . .
. . . .
. Q . .

Q . . .
. . Q .
. . . .
. Q . .

. . . .
Q . . .
. . . .
. . . .
```

```
CSP_NQueens x
↑ Q . . .
↓ . . Q .
≡
. . . .
. . . .
. . . .
Q . . .

. Q . .
. . . .
. . . .
Q . . .

. Q . .
. . . .
. . Q .
Q . . .

. . . .
. Q . .
. . . .
Q . . .

Process finished with exit code 0
```