**Name : Ali Azhar**

**Roll Number : SU92-BSAIM-S24-026**

**Section : 4A**

**“Documentation of Task 4”**

**N-Queens Problem (Backtracking)**

**Purpose:**  
This program solves the **N-Queens problem** using **backtracking**.  
The task: Place N queens on an N x N chessboard so that no two queens attack each other.

### What the code does:

1. **print\_board(board, N)**
   * Prints the chessboard.
   * Q = queen, . = empty square.
2. **is\_safe(board, row, col, N)**
   * Checks if a queen can be safely placed at (row, col).
   * Ensures no queen exists in the same column, and both diagonals.
3. **solve(board, row, N)**
   * Recursive function.
   * Tries placing a queen row by row.
   * If valid, moves to the next row.
   * If a full solution is found, prints the board.
4. **n\_queens(N)**
   * Creates an empty board.
   * Calls solve() to find solutions.
   * If no solution exists, prints a message.