***Lab Task 4***

***Name :*** M. Hassan   
***Roll no. :***101

***Report: Solving the N-Queens Problem Using Backtracking***

## **1. Introduction**

The N-Queens Problem is a classic combinatorial problem in which the goal is to place N queens on an N×N chessboard such that no two queens attack each other. This report presents an implementation using a backtracking algorithm and visualizes the solutions using Matplotlib.

## **2. Problem Statement**

* Place N queens on an N×N chessboard.
* Ensure that no two queens share the same row, column, or diagonal.
* Generate and visualize all valid solutions.

## **3. Methodology**

### **3.1 Approach**

* A recursive backtracking function (solve\_n\_queens) explores all possible placements.
* A helper function (is\_safe) checks if a queen can be placed in a given position.
* Solutions are stored in a list and displayed using both text-based formatting and graphical representation.

### **3.2 Implementation**

* The algorithm iterates through columns in each row and places a queen if the position is valid.
* If all rows are successfully filled, a solution is recorded.
* The display\_board function visualizes solutions using Matplotlib.
* The program prompts the user for input and handles invalid values gracefully.

## **4. Results and Conclusion**

* The program successfully finds and displays all valid solutions for a given N.
* The backtracking approach ensures an efficient search for solutions.
* Future improvements could include optimizing the algorithm using constraint propagation or heuristic-based approaches.

**End of Report**

**Screenshots**

