PHÂN TÍCH VÀ THIẾT KẾ THUẬT TOÁN

BACKTRACKING

**Practice problem 1:**

**Đề bài**: **N-Queen Problem:**

The N Queen is the problem of placing **N** chess queens on an **N×N** chessboard so that no two queens attack each other.

Given an integer n, return all distinct solutions to the n-queens puzzle. You may return the answer in any order Each solution contains a distinct board configuration of the n-queens’ placement, where ‘Q’ and ‘.’ Both indicate a queen and an empty space, respectively.

**Trả lời:**

*The idea is to place queens one by one in different columns, starting from the leftmost column. When we place a queen in a column, we check for clashes with already placed queens. In the current column, if we find a row for which there is no clash, we mark this row and column as part of the solution. If we do not find such a row due to clashes, then we backtrack and return false.*

* Start in the leftmost column
* If all queens are placed return true
* Try all rows in the current column. Do the following for every tried row.
  + If the queen can be placed safely in this row
    - Then mark this [row, column] as part of the solution and recursively check if placing queen here leads to a solution.
    - If placing the queen in [row, column] leads to a solution then return true.
    - If placing queen doesn’t lead to a solution then unmark this [row, column] and track back and try other rows.
  + If all rows have been tried and nothing worked return false to trigger backtracking.