**N-QUEENS PROBLEMS**

**Introduction:**

The N-Queens problem is a classic puzzle in which the goal is to place N queens on an N x N chessboard such that no two queens threaten each other. This means no two queens can share the same row, column, or diagonal. The problem is a great way to practice and understand backtracking algorithms.

**Approach:**

**Initialization:**

1. Start with an empty chessboard.
2. The board is represented by a 2D list where each cell is initially set to 0.

**Place Queens:**

1. Begin by placing the first queen in the first row and first column.
2. Move to the next row and try placing a queen in each column one by one.

**Check Safety**:

1. Before placing a queen in a cell, check if it is safe.
2. A cell is considered safe if no other queens are present in the same column, or on the diagonals.

**Backtrack**:

1. If placing the queen in the current cell leads to a solution, continue to place the next queen.

2. If placing the queen in the current cell does not lead to a solution, remove the queen (backtrack) and try the next column.

**Python Code for the N-Queens Problems:**

def is\_safe(board, row, col, N):

# Check this column on the upper side

for i in range(row):

if board[i][col] == 1:

return False

# Check the upper left diagonal

for i, j in zip(range(row, -1, -1), range(col, -1, -1)):

if board[i][j] == 1:

return False

# Check the upper right diagonal

for i, j in zip(range(row, -1, -1), range(col, N)):

if board[i][j] == 1:

return False

return True

def solve\_n\_queens\_util(board, row, N):

if row >= N:

return True

for col in range(N):

if is\_safe(board, row, col, N):

board[row][col] = 1 # Place the queen

if solve\_n\_queens\_util(board, row + 1, N):

return True

board[row][col] = 0 # Backtrack

return False

def solve\_n\_queens(N):

board = [[0] \* N for \_ in range(N)]

if not solve\_n\_queens\_util(board, 0, N):

print("Solution does not exist")

return False

# Print the solution

for row in board:

print(" ".join(str(cell) for cell in row))

return True

**Example:**

Let's solve the problem for N = 4. Here's a possible solution:

Board:

0 1 0 0

0 0 0 1

1 0 0 0

0 0 1 0

In this solution:

* The first queen is placed at (0,1).
* The second queen is placed at (1,3).
* The third queen is placed at (2,0).
* The fourth queen is placed at (3,2).