8 QUEEN PROBLEM

SOURCE CODE:

```
N = 8

def printSolution(board):
    for row in board:
        for i in range(N):
            print("Q" if row[i] else ".", end=" ")
        print()
    print()  # Add a newline for readability

def isSafe(board, row, col):
    # Check column
    for i in range(row):
        if board[i][col]:
            return False

    for i, j in zip(range(row - 1, -1, -1), range(col - 1, -1, -1)):
        if board[i][j]:
            return False


    for i, j in zip(range(row - 1, -1, -1), range(col + 1, N)):
        if board[i][j]:
            return False

    return True

def solve(board, row):
    if row == N:
        print("One possible solution is:")
        printSolution(board)
        return True  # Stop after finding one solution

    for col in range(N):
        if isSafe(board, row, col):
            board[row][col] = 1  # Place queen

            if solve(board, row + 1):
                return True  # Stop after the first valid solution

            board[row][col] = 0  # Backtrack
    return False

def eightQueens():
    board = [[0 for _ in range(N)] for _ in range(N)]
    if not solve(board, 0):
        print("Solution does not exist")
eightQueens()
```

OUTPUT: [redacted] One possible solution is:

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