# N-QUEENS PROBLEM.

AIM :-

To solve the n-Queens problem where the goal is to place n-queens on a n×n chessboard such that no two queens attack each other.

ALGORITHM :-

Step 1 : Start.

Step 2 : Create a n×n chessboard with all chess cells to 0, representing no queens placed.

Step 3 :- Move to the next col if placing a queen works, else backtrack by removing queen.

Step 4 : Display the board.

Step 5 : If no sol exist, print "solution does not exists.

PROGRAM :-

```
def isSafe(board, row, col, n):
    for i in range(col):
        if board[row][i] == 1
            return false :
for i, j in zip(range(row, -1, -1), range
                                (col, -1, -1)):
        if board[i][j] == 1 :
            return false
    return true.
```

```
def solve NQueen(board, col, n):
    if col >= n:
        return true
    for i in range(n):
        if isSafe(board, i, col, n):
            board[i][col] = 1.
        if solve NQ util(board, col+1, n) == true.
            return true.
        board[i][col] = 0.
    return false.
def solve NQ(n):
    board = [[0]*n for _ in range(n)]
    if solveNQutil(board, 0, n) == false
        print("Solution does not exist").
    return false.
for i in board:
    print(i)
    return true.
n = int(input("enter n value : ").
solve NQ(n).
```

output :-

Enter value : 5

[1, 0, 0, 0, 0]

[0, 0, 0, 1, 0]

[0, 1, 0, 0, 0]

[0, 0, 0, 0, 1] → [0, 0, 1, 0, 0]

result :-

The program is successfully executed & output is verified -