**DAA Assignment 5 N-Queens Problem**

**Code:**

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

    for i in range(row):

        if board[i][col] == 1:

            return False

    i, j = row - 1, col - 1

    while i >= 0 and j >= 0:

        if board[i][j] == 1:

            return False

        i -= 1

        j -= 1

    i, j = row - 1, col + 1

    while i >= 0 and j < n:

        if board[i][j] == 1:

            return False

        i -= 1

        j += 1

    return True

def place\_queens(board, row, n):

    if row == n:

        yield [row[:] for row in board]

    else:

        for col in range(n):

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

                board[row][col] = 1

                yield from place\_queens(board, row + 1, n)

                board[row][col] = 0

def print\_board(board, n):

    for i in range(n):

        for j in range(n):

            print(board[i][j], end=" ")

        print()

def main():

    n = int(input("Enter the number of queens: "))

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

    solutions = list(place\_queens(board, 0, n))

    if solutions:

        print(f"Found {len(solutions)} solutions:")

        for i, solution in enumerate(solutions):

            print(f"Solution {i+1}:")

            print\_board(solution, n)

    else:

        print("No solution exists")

if \_\_name\_\_ == "\_\_main\_\_":

    main()

**Output:**

Enter the number of queens: 4

Found 2 solutions:

Solution 1:

0 1 0 0

0 0 0 1

1 0 0 0

0 0 1 0

Solution 2:

0 0 1 0

1 0 0 0

0 0 0 1

0 1 0 0