**Assignment No. 4**

**Name : Sonu Shriram Vishwakarma**

**Roll No. : TAI&D61 (TAI&D32 new)**

**Subject: Artificial Intelligence (AI)**

**Aim:**

Implement a solution for a constraint satisfaction problem using branch and bound and backtracking for N-queen Problem.

**Code :**

# Coded By: Sonu Shriram Vishwakarma

cnt = 0

def isSafe(board, n, row, col):

    # To store row and col postition in temperory variables so that we can access it later

    newrow = row

    newcol = col

    # To check uppper diagonal

    while (row >= 0 and col >= 0):

        if(board[row][col] == 1):

            return False;

        row -= 1

        col -= 1

    col = newcol;

    row = newrow;

    # To check left horizontal

    while(col >= 0):

        if(board[row][col] == 1):

            return False;

        col -= 1

    row = newrow;

    col = newcol;

    # To check lower left diagonal

    while (row<n and col>=0):

        if(board[row][col] == 1):

            return False;

        row += 1

        col -= 1

    return True;

def solveBoard(board, n, col):

    global cnt

    if(col == n):

        cnt +=1

        print(f"Solution No. {cnt}: ", board)

        return

    # Tranverse all the rows and try to place queen in each column using recursion call

    for row in range(n):

        if(isSafe(board, n, row, col)):

            board[row][col] = 1

            solveBoard(board, n, col+1)

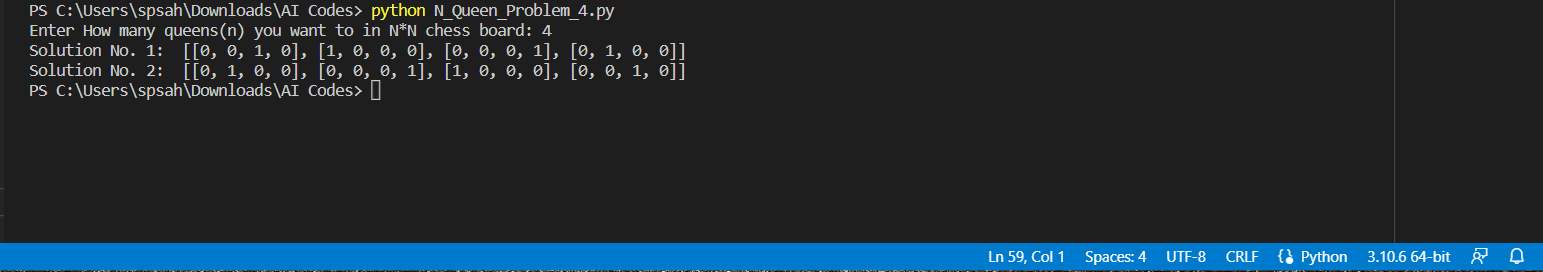
            board[row][col] = 0

n = int(input("Enter How many queens(n) you want to in N\*N chess board: "))

board = [[0 for j in range(n)] for i in range(n)]

solveBoard(board, n, 0)

**Output :**

****