

Name: Ashutosh Raj Gupta

Roll no- TECO2223A045

Assignment no: 4

Implement a solution for a Constraint Satisfaction Problem using Branch and Bound and Backtracking for n-queens problem or a graph coloring problem.

Code:

```
N = int(input("Enter no of queens: "))

# for printing boards
board = [[0]*N for _ in range(N)]

def attack(i, j):

    for k in range(0, N):
        if board[i][k] == 1 or board[k][j] == 1:
            return True

    for k in range(0, N):
        for l in range(0, N):
            if (k+l == i+j) or (k-l == i-j):
                if board[k][l] == 1:
                    return True
    return False

def N_queens(n):
    if n == 0:
        return True
    for i in range(0, N):
        for j in range(0, N):
            if (not (attack(i, j))) and (board[i][j] != 1):
                board[i][j] = 1
                if N_queens(n-1) == True:
                    return True
                board[i][j] = 0
    return False

N_queens(N)
for i in board:
    print(i)
```

output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  COMMENTS

● PS C:\Users\Ashutosh Raj Gupta\Desktop\sem6 Laboratory\LP2\LP2-Assignments\AI> python -u "c:\Users\Ashutosh Raj Gupta\Desktop\sem6
AI\Assignment4\NQUEEN.py"
● Enter no of queens: 4
[0, 1, 0, 0]
[0, 0, 0, 1]
[1, 0, 0, 0]
[0, 0, 1, 0]
PS C:\Users\Ashutosh Raj Gupta\Desktop\sem6 Laboratory\LP2\LP2-Assignments\AI> python -u "c:\Users\Ashutosh Raj Gupta\Desktop\sem6
AI\Assignment4\NQUEEN.py"
Enter no of queens: 8
[1, 0, 0, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 1, 0, 0, 0]
[0, 0, 0, 0, 0, 0, 0, 1]
[0, 0, 0, 0, 0, 1, 0, 0]
[0, 0, 1, 0, 0, 0, 0, 0]
[0, 0, 0, 0, 0, 0, 1, 0]
[0, 1, 0, 0, 0, 0, 0, 0]
[0, 0, 0, 1, 0, 0, 0, 0]
○ PS C:\Users\Ashutosh Raj Gupta\Desktop\sem6 Laboratory\LP2\LP2-Assignments\AI> 
```