

EXPI

N QUEENS.

AIM :- To write the program for N-QUEENS problem and execute successfully.

PSEUDOCODE:-

1. Function n_queens(N):

2. CREATE a board of size $N \times N$ filled with 0

3. If solve_n_queens(board, 0, N) is TRUE:

4. PRINT the board

5. ELSE:

6. PRINT "Solution does not exist"

7. Function solve_n_queen(board, col, N):

8. If col \geq N:

9. Return True

10. For each row from 0 to N-1:

11. If is_safe(board, row, col, N) is TRUE:

12. PLACE Queen at (row, col)

13. If solve_n_queen(board, col+1, N) is True:

14. RETURN TRUE

15. REMOVE queen from (row, col) (Backtrack)

16. RETURN FALSE.

17. FUNCTION is_safe(board, row, col, N):

18. FOR each column left of (row, col):

19. CHECK row, upper diagonal and lower diagonal.

20. If any queen is found in these directions:

21. RETURN FALSE

22. RETURN TRUE

PROGRAM :-

```
def print_board(board):
    for row in board:
        print(" ".join(str(x) for x in row))
    print("\n")

def is_safe(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

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

    return True

def solve_n_queen(board, col, n):
    if col >= n:
        return True

    for i in range(n):
        if is_safe(board, i, col, n):
            board[i][col] = 1

            if solve_n_queens(board, col+1, n):
                return True

            board[i][col] = 0

    return False

def n_queens(n):
    board = [[0]*n for _ in range(n)]
    if not solve_n_queens(board, 0, n):
        print("Solution does not exist")
        return False
```

print_board (board)

return True

try:

n = int (input ("Enter the value of N (size of the board):"))

if $n \leq 0$:

print ("Please enter a positive integer.")

else:

n_queens(n)

except ValueError:

print ("Invalid input! Please enter a valid integer.")

OUTPUT :-

Enter the value of N (size of the board): 4

0 0 0 1

1 0 0 0

0 0 0 1

0 1 0 0

RESULT:-

Therefore, the program is successfully executed and output is verified.