Name :

Awais Ahmad

Roll-No:

SU92-BSAIM-F23-132

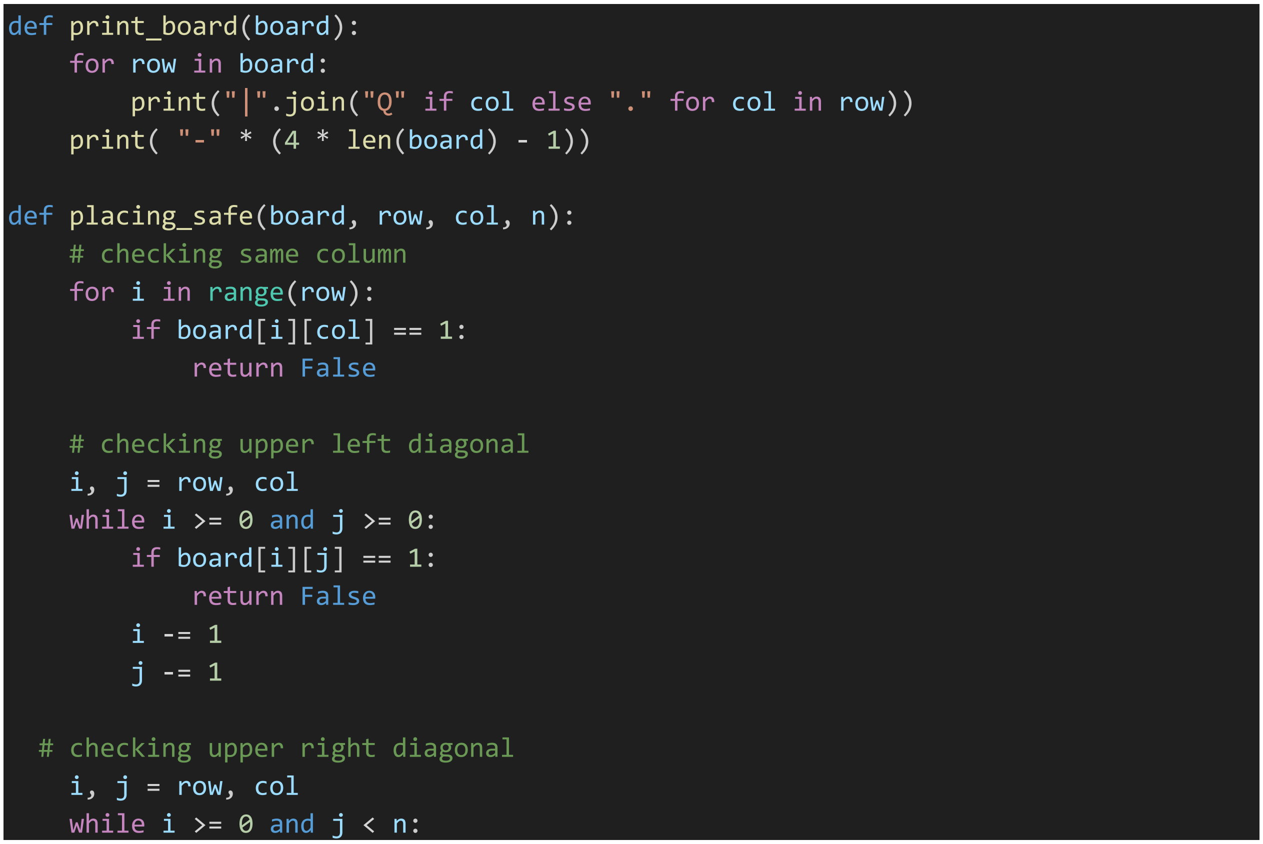
Section:

BSAI-4C

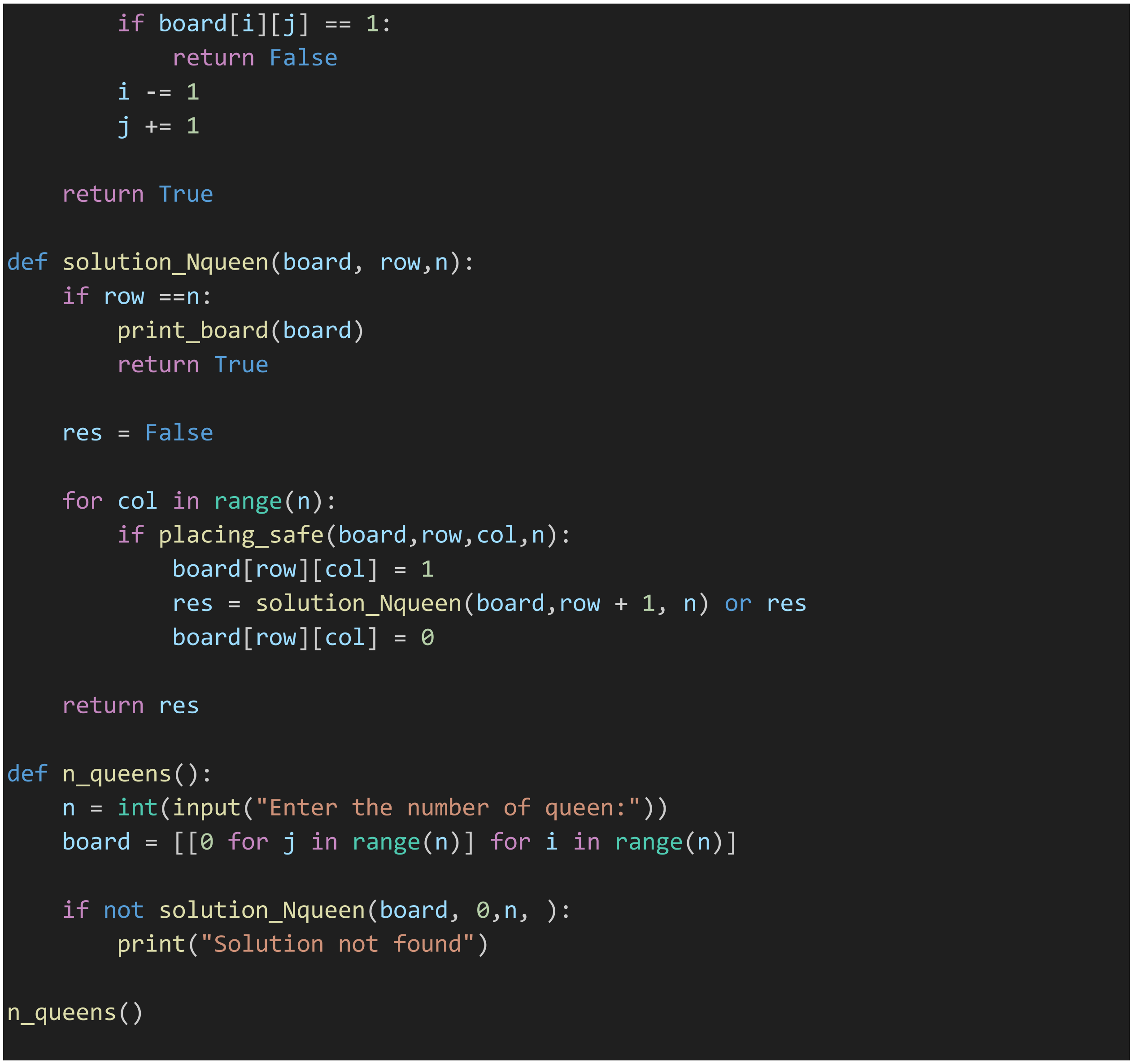
Task:

(4)

Task : N-Queens Problem (Dynamic)

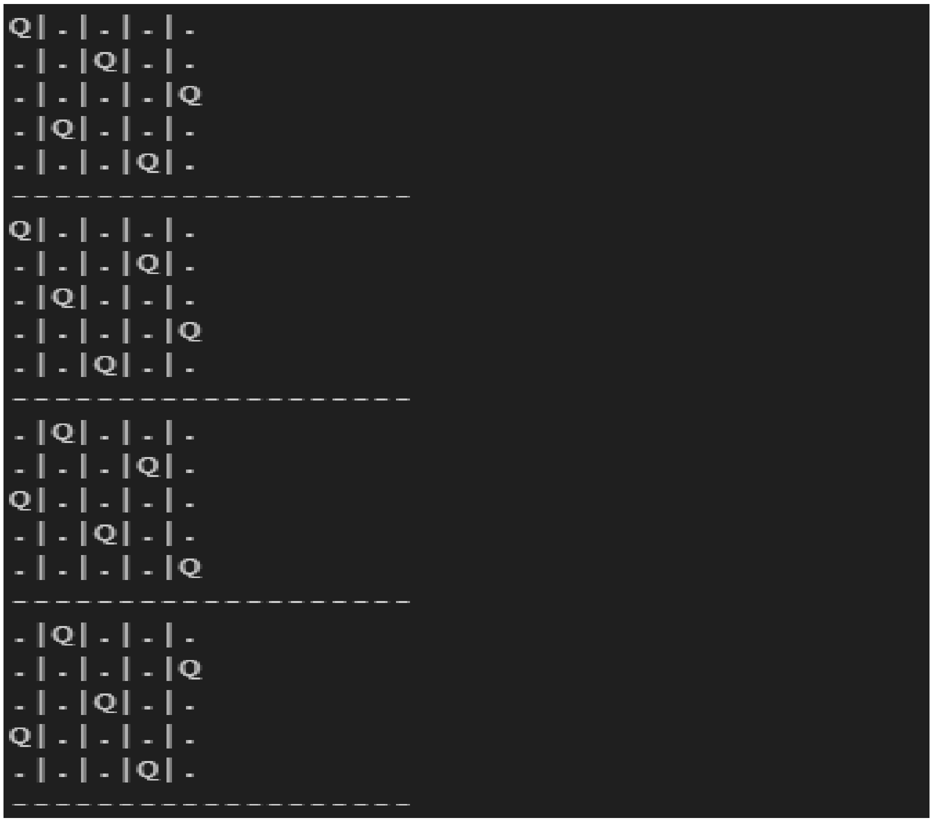


```
def print_board(board):
    for row in board:
        print("|".join("Q" if col else "." for col in row))
    print( "-" * (4 * len(board) - 1))

def placing_safe(board, row, col, n):
    # checking same column
    for i in range(row):
        if board[i][col] == 1:
            return False

    # checking upper left diagonal
    i, j = row, col
    while i >= 0 and j >= 0:
        if board[i][j] == 1:
            return False
        i -= 1
        j -= 1

  # checking upper right diagonal
    i, j = row, col
    while i >= 0 and j < n:
```



```python
        if board[i][j] == 1:
            return False
        i -= 1
        j += 1

    return True

def solution_Nqueen(board, row,n):
    if row ==n:
        print_board(board)
        return True

    res = False

    for col in range(n):
        if placing_safe(board,row,col,n):
            board[row][col] = 1
            res = solution_Nqueen(board,row + 1, n) or res
            board[row][col] = 0

    return res

def n_queens():
    n = int(input("Enter the number of queen:"))
    board = [[0 for j in range(n)] for i in range(n)]

    if not solution_Nqueen(board, 0,n, ):
        print("Solution not found")

n_queens()
```

Output:



```
Q| . | . | . | .
 . | . |Q| . | .
 . | . | . | . |Q
 . |Q| . | . | .
 . | . | . |Q| .
-------------------
Q| . | . | . | .
 . | . | . |Q| .
 . |Q| . | . | .
 . | . | . | . |Q
 . | . |Q| . | .
-------------------
 . |Q| . | . | .
 . | . | . |Q| .
Q| . | . | . | .
 . | . |Q| . | .
 . | . | . | . |Q
-------------------
 . |Q| . | . | .
 . | . | . | . |Q
 . | . |Q| . | .
Q| . | . | . | .
 . | . | . |Q| .
-------------------
```