Name: Piyush Chaudhari

Roll No: 20U419

Class: T.E. Comp

Div: 4

Batch: T14

Assignment 4

class Solution:

def solveNQueens(self, n: int):

col = set()

posDiag = set() # r+c

negDiag = set() # r-c

res = []

board = [['0'] \* n for i in range(n)]

def backtrack(r): # row

if r == n: # means that we were able to find a valid solution

copy = ["".join(row) for row in board]

res.append(copy)

return

for c in range(n):

if c in col or r + c in posDiag or r - c in negDiag: # These place cannot be used

continue

# as the above place cannot be used so add it to the set

col.add(c)

posDiag.add(r + c)

negDiag.add(r - c)

board[r][c] = "1" # update the board

backtrack(r + 1)

col.remove(c)

posDiag.remove(r + c)

negDiag.remove(r - c)

board[r][c] = "0" # after backtracking

backtrack(0)

for i in res:

print(i)

s = Solution()

s.solveNQueens(4)

output:

