class Solution(object):

def generateParenthesis(self, n):

res = []

self.dfs(n, n, "", res)

return res

def dfs(self, leftRemain, rightRemain, path, res):

if leftRemain > rightRemain or leftRemain < 0 or rightRemain < 0:

return # backtracking

if leftRemain == 0 and rightRemain == 0:

res.append(path)

return

self.dfs(leftRemain-1, rightRemain, path+"(", res)

self.dfs(leftRemain, rightRemain-1, path+")", res)