

```

class Solution:
    def removeInvalidParentheses(self, s: str) -> List[str]:

        left, right = 0, 0
        for c in s:
            if c == "(": left += 1
            if c == ")":
                right = right + 1 if left <= 0 else right
                left = left - 1 if left > 0 else left

        ans = set()
        leftR, rightR = left, right

        def backtrack(leftR, rightR, left, right, expr: List[str], idx):

            if idx == len(s):
                if leftR == rightR == 0:
                    ans.add("".join(expr))
                return

            c = s[idx]

            if c not in "()":
                backtrack(leftR, rightR, left, right, expr+[c], idx+1)

            else:
                if (c == "(" and leftR>0) or (c == ")" and rightR>0): # skip
                    backtrack(leftR - (c=="("), rightR - (c==")"), left,
right, expr, idx+1)
                    # add c

                if c == "(":
                    backtrack(leftR, rightR, left+1, right, expr+[c], idx+1)

                if c == ")" and left>0:
                    backtrack(leftR, rightR, left-1, right, expr+[c], idx+1)

        backtrack(leftR, rightR, 0,0, [], 0)
        return ans

```