Python Program for Water Jug Problem

from collections import deque

def water\_jug(x, y, target):

q = deque()

q.append((0, 0, [])) # (jugX, jugY, steps taken)

visited = set()

while q:

a, b, steps = q.popleft()

if (a, b) in visited:

continue

visited.add((a, b))

# check if target reached

if a == target or b == target:

for s in steps + [(a, b)]:

print("JugX =", s[0], "JugY =", s[1])

return

# all possible moves

moves = [

(x, b, "Fill X"), (a, y, "Fill Y"),

(0, b, "Empty X"), (a, 0, "Empty Y"),

(a - min(a, y - b), b + min(a, y - b), "Pour X->Y"),

(a + min(b, x - a), b - min(b, x - a), "Pour Y->X")

]

for nx, ny, \_ in moves:

q.append((nx, ny, steps + [(a, b)]))

x, y, target = 4, 3, 2 # Jug capacities & target

water\_jug(x, y, target)

Output

JugX = 0 JugY = 0

JugX = 0 JugY = 3

JugX = 3 JugY = 0

JugX = 3 JugY = 3

JugX = 4 JugY = 2