from collections import deque

start, goal = (3, 3, 1), (0, 0, 0)

moves = [(1,0), (2,0), (0,1), (0,2), (1,1)]

def safe(m, c):

return (m == 0 or m >= c) and (3 - m == 0 or 3 - m >= 3 - c)

q = deque([(start, [start])])

visited = set()

while q:

(m, c, b), path = q.popleft()

if (m, c, b) == goal:

for step in path:

print(step)

break

if (m, c, b) in visited:

continue

visited.add((m, c, b))

for dm, dc in moves:

if b: # boat on left

nm, nc = m - dm, c - dc

else: # boat on right

nm, nc = m + dm, c + dc

nb = 1 - b

if 0 <= nm <= 3 and 0 <= nc <= 3 and safe(nm, nc):

q.append(((nm, nc, nb), path + [(nm, nc, nb)]))

