WATER JUG PROBLEM

AIM: You are given two jugs, a 4-gallon one and a 3-gallon one, a pump which has unlimited water which you can use to fill the jug, and the ground on which water may be poured. Neither jug has any measuring markings on it. How can you get exactly 2 gallons of water in the 4-gallon jug?

PROCEDURE:

So, to solve this problem, following set of rules were proposed:

Production rules for solving the water jug problem

Here, let ***x*** denote the 4-gallon jug and ***y*** denote the 3-gallon jug.

S.No.	Initial State	Condition	Final state	Description of action taken
1.	(x,y)	If x<4	(4,y)	Fill the 4 gallon jug completely
2.	(x,y)	if y<3	(x,3)	Fill the 3 gallon jug completely
3.	(x,y)	If x>0	(x-d,y)	Pour some part from the 4 gallon jug
4.	(x,y)	If y>0	(x,y-d)	Pour some part from the 3 gallon jug
5.	(x,y)	If x>0	(0,y)	Empty the 4 gallon jug
6.	(x,y)	If y>0	(x,0)	Empty the 3 gallon jug
7.	(x,y)	If (x+y)<7	(4, y-[4-x])	Pour some water from the 3 gallon jug to fill the four gallon jug
8.	(x,y)	If (x+y)<7	(x-[3-y],y)	Pour some water from the 4 gallon jug to fill the 3 gallon jug.
9.	(x,y)	If (x+y)<4	(x+y,0)	Pour all water from 3 gallon jug to the 4 gallon jug
10.	(x,y)	if (x+y)<3	(0, x+y)	Pour all water from the 4 gallon jug to the 3 gallon jug

The listed production rules contain all the actions that could be performed by the agent in transferring the contents of jugs.

But, to solve the water jug problem in a minimum number of moves, following set of rules in the given sequence should be performed:

Solution of water jug problem according to the production rules:

S.No.	4 gallon jug contents	3 gallon jug contents	Rule followed
1.	0 gallon	0 gallon	Initial state
2.	0 gallon	3 gallons	Rule no.2
3.	3 gallons	0 gallon	Rule no. 9
4.	3 gallons	3 gallons	Rule no. 2
5.	4 gallons	2 gallons	Rule no. 7
6.	0 gallon	2 gallons	Rule no. 5
7.	2 gallons	0 gallon	Rule no. 9

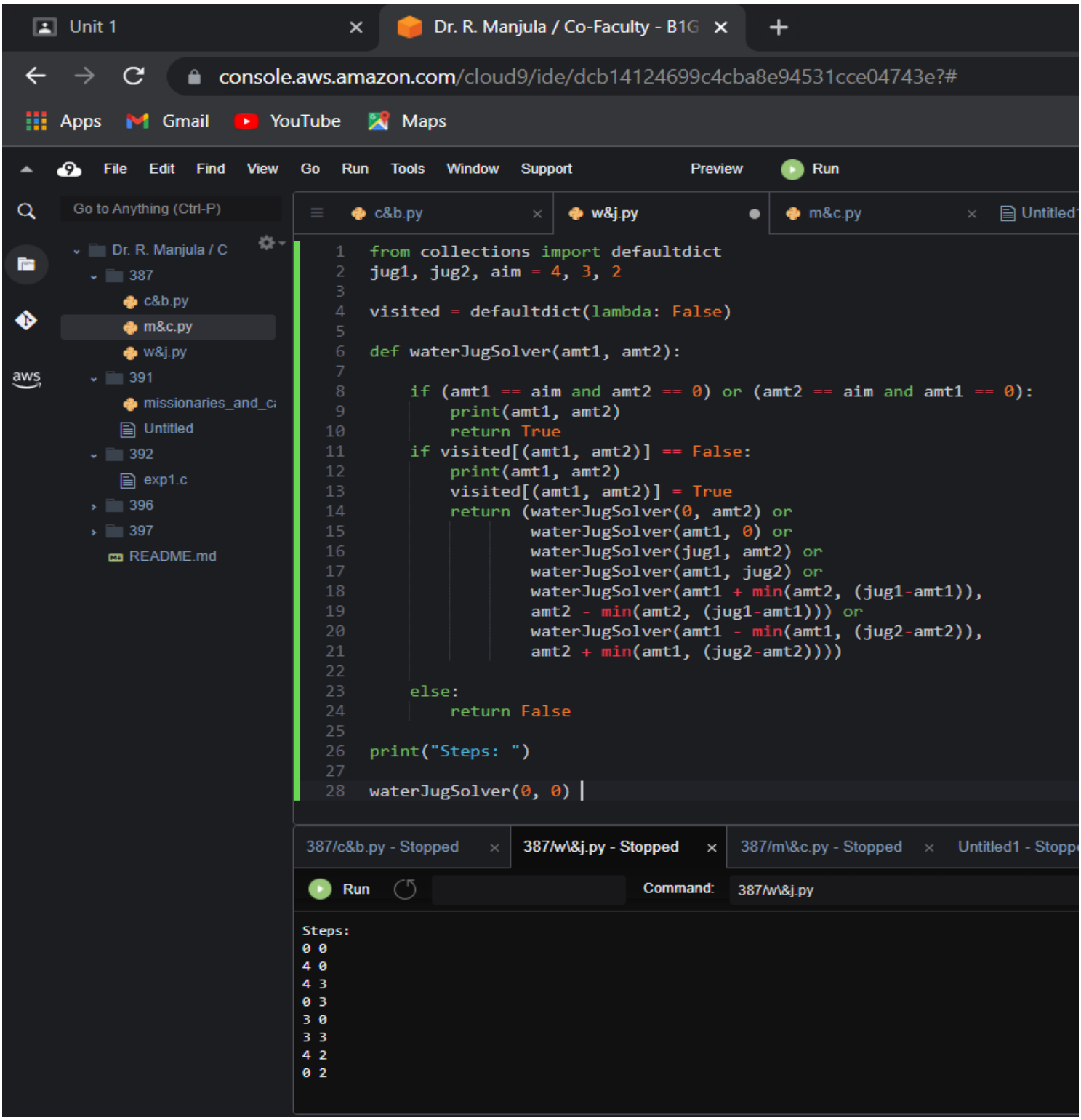
On reaching the 7^{th} attempt, we reach a state which is our goal state. Therefore, at this state, our problem is solved.

CODE:

```
from collections import defaultdict

jug1, jug2, aim = 4, 3, 2


visited = defaultdict(lambda: False)


def waterJugSolver(amt1, amt2):


    if (amt1 == aim and amt2 == 0) or (amt2 == aim and amt1 ==
0):
```

```
        print(amt1, amt2)
        return True
    if visited[(amt1, amt2)] == False:
        print(amt1, amt2)
        visited[(amt1, amt2)] = True
        return (waterJugSolver(0, amt2) or
                waterJugSolver(amt1, 0) or
                waterJugSolver(jug1, amt2) or
                waterJugSolver(amt1, jug2) or
                waterJugSolver(amt1 + min(amt2, (jug1-amt1)),
                amt2 - min(amt2, (jug1-amt1))) or
                waterJugSolver(amt1 - min(amt1, (jug2-amt2)),
                amt2 + min(amt1, (jug2-amt2))))

    else:
        return False
print("Steps: ")
waterJugSolver(0,0)
```

OUTPUT:


Unit 1
Dr. R. Manjula / Co-Faculty - B1G
console.aws.amazon.com/cloud9/ide/dcb14124699c4cba8e94531cce04743e?#
Apps
Gmail
YouTube
Maps
File Edit Find View Go Run Tools Window Support
Preview
Run
Go to Anything (Ctrl-P)
Dr. R. Manjula / C
387
c&b.py
m&c.py
w&j.py
391
missionaries_and_c
Untitled
392
exp1.c
396
397
README.md
c&b.py
w&j.py
m&c.py
Untitled
1 from collections import defaultdict
2 jug1, jug2, aim = 4, 3, 2
3
4 visited = defaultdict(lambda: False)
5
6 def waterJugSolver(amt1, amt2):
7
8 if (amt1 == aim and amt2 == 0) or (amt2 == aim and amt1 == 0):
9 print(amt1, amt2)
10 return True
11 if visited[(amt1, amt2)] == False:
12 print(amt1, amt2)
13 visited[(amt1, amt2)] = True
14 return (waterJugSolver(0, amt2) or
15 waterJugSolver(amt1, 0) or
16 waterJugSolver(jug1, amt2) or
17 waterJugSolver(amt1, jug2) or
18 waterJugSolver(amt1 + min(amt2, (jug1-amt1)),
19 amt2 - min(amt2, (jug1-amt1))) or
20 waterJugSolver(amt1 - min(amt1, (jug2-amt2)),
21 amt2 + min(amt1, (jug2-amt2))))
22
23 else:
24 return False
25
26 print("Steps: ")
27
28 waterJugSolver(0, 0)
387/c&b.py - Stopped
387/w\&j.py - Stopped
387/m\&c.py - Stopped
Untitled1 - Stopp
Run
Command:
387/w\&j.py
Steps:
0 0
4 0
4 3
0 3
3 0
3 3
4 2
0 2

RESULT: **The water jug problem was successfully executed**

RA1911003010387

CHITRALEKHA.CH