

Exp. No. 3

M	T	W	T	F	S	S
Page No.:						YOUVA
Date:						

Topic:

Water jug problem using DFS.

Aim:

To determine if a specific amount of water can be measured using two jugs of different capacities via DFS.

Algorithm:

Step 1: Start.

Step 2: Rep. each state as a tuple.

Step 3: Implement DFS to explore all states.

Step 4: Use a set to track visited states.

Step 5: Generate two new states by filling, emptying, or transferring.

Step 6: Return true if the target volume is reached.

Step 7: Apply DFS recursively to each new state.

Step 8: Explore all possible states.

Step 9: Step if goal reached.

Program:

def wjd (j_1, j_2, t):

def dfs (state):

if state in visited:

return false.

visited = false

visited.add (state)

$j_1, j_2 = state$

if $j_1 == target$ or $j_2 == target$:

return true.

actions = [(j_1, j_2),

($j_1, j_2 - c$),

(0, j_2),

($j_1, 0$)

($\min(j_1 - c, j_1 + j_2)$

$j_2 - (\min(j_2 - c, j_1 + j_2 - j_1)$

($j_1 - (\min(j_2 - c, j_1 + j_2 - j_2)$

$- j_2) - \min(j_2 - c, j_1 + j_2)$

return any (dfs (new state)) for new state in action

visited = set()

return dfs ((0, 0))

$j_1 - c = 4$

$j_2 - c = 3$

target = 2

print (wjd ($j_1 - c, j_2 - c, target$))

Output :

True

Result:

Thus the program is successful executed &

is verified

①