Exporment No:3	DFS	(Water	Jug	PROBLEM)
----------------	-----	--------	-----	----------

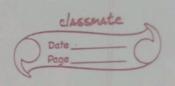
	C	5			
A	1		Y	٦	٨
73	•	•		•	-

To solve the water Jug Problem using the Depth - Fast seasch (DFS) algorithm. The goal is to measure exactly 2 litres of water in the 4 liter jug while ensuring that the 3-liter jug is empty, using a series of operations like filling emptyeng and pouring between the two jugs.

Algorithm:

- State Representation: Represent the state of the jugs as a tuple (n,y) where n is the amount of water En the 4 liter jug and y is the amount of water in the 3-liter jug.
- 2) Initial State: Stoat with both jugs empty: (0,0).
- Défine Possible operations:
 - · Fill the 4- liter jug: (4,4)
 - Fill the 3- liter jug: (713)
 - · Empty the 4-liter jug: (0,4)
 · Empty the 3-liter jug: (11,0)

 - Jug until one is empty of the other is full: (x-min(y,4-x), y-min (y,4-x)).
 - . Pous water grom the 3-liter jug to the 4-liter jug until one is empty or the other is full: (x+min(y,4-x), y-min (y, 4-x)).
- use DFS to Explose States:
 - · Start from the initial state (0,0) and use DFS to explose all possible states by applying the operations. · Moss each visited state to avoid sevisiting.
 - · IP the state (2,0) is reached, where the 4-liter jug lias 2 lites and the 3-lites jug is empty. He solution is found.



Backtracking: Ip a state leads to me justing valid states, backtrack and try a different operation. of is valid. state (next or next y) and Paggam: (alog, hallely, y hear in Axor) as Sultan TRUE del 28-valid-state(n,y): # Ensure that the state is within the capacity of the jugs . (1909. Stor 2etuan 0 <= x <= 4 and 0 <= y <= 3 # mask the current state as visited visited. add ((21.41) . dd (2) + . . . # stole the current state in the path path. append (my)) # IR we have 2 leters of water in 4 liter jug and o lites in 3 lites jug, the problem is 28 n==2 and y==0: " = 9 mon se solved. Rotuan True ma Idang pul so four 5000x # possible moves possible_moves = [(4,4), # Fill the 4-liter jug (0,y), # Empty the 4-liter jug (7,0), # Empty the 3- liter gug (n-min (n,3-y), y+min (n,3-y)), # Pous from 4L to3L (2+min (y, 4-21), y-min (y, 4-21)) # pous from 31 to 41

Explore all possible moves using DFS
pos (mext_or, next_y) in possible_moves: of is-valid-state (next-n, nent-y) and des (next_x, next-y, visited, poth): setuan True # Backtrack: remove the last state of it leads to no solution and the path.popl) zetian False has person and de solve-water-jug-problem (): initial_state= (0,0)/2000 000 (pand se visited = get () path = [] of als linitial state [0], initial state [1], visited, path). print (" Solution gound!"). has the step in path: 00 201 2802 14 print (step) (gold) broggo Alon elle: print ("NO solution exists") and o steem in a cities jug itee problem is # Main code % _name_== "__main__" : bon come so solve_water jug_problem () of realiste moves Dessible moves : [(4.4) (# Fill the 4- cetes jug (21.3) # Fill the 8+ cetes jug (04) It Empty the 4-19th Pac (o, c), is toply the 3- lites fug (or an in (or 3-4), by + on in (or, 3-4)), # Peus pen 41 fest (2 + mile (4, 4 2) y mile (9, 4 - 20) it pour from st to 16

-		
D	ate	7
Po	ige	

	Page
	output: As the su thanksque
	solution found!
	(0,0)
Python.	
legt !	13,0) 63 31 body 25 million *A 031
n a	9 ho (3,3) se a si sobre tente a mara stor
180-18	(4,2) ten south of the police states
	la (0,2) et elomites siteres à bus shou.
	(2,0)
	m.Bl9Rgpj A
.0	of tritalize the graph represent the graph as
tuples	distributy where each made has a list of
	अवश्याद्वार है कि नाविधिकारिक नाविधि वार्त महि
	te seach them.
(0)30	a) Hearstic Frotien Deping a leasistic function that estimates the cost from mode in the the
	mode. The charge of housethe depends on the
- COV	preblem demand. Fer sampleaty assume we are g
	His Censistic Junetion
	3) DEKNE FLE A SEONCH FLOCKENS
(dualat)	Tribialize but sets : epan set (mortes to be e
	and elect. set (nodes already evaluated).
	्थार व प्रशिक्षित वार्षाय (कार्ती- रिव्वन) यह प्रारक्षित के
s His	with the seminat & (n) = 9(n) + 6(n), ruhase 9(n)
اثو	cest from the stast made to m and folm is
	Exersistic estimate from n is the goal
	Trittalize the g valides (usts grown the start)
en Elich	o hom Starte such secure tille starte mode
	Result: 0 of basillabling 20
60,8	Thus, the program for Water Jug Roblem
	as successfully executed and the output is verified
4:	etaner the posent of ench mode to seemster
	ties pate sonching the goal