

**FUNDAMENTALS OF ARTIFICIAL
INTELLIGENCE
Lab Exercise – 3**

**Name : ADVAIT GURUNATH CHAVAN
Email ID : advaitchavan135@gmail.com**

**CDAC – NOIDA
PGDAI**

ASSIGNMENT NO 3

FOR EVERY ANSWER MENTION THE NUMBER OF NODES EXPANDED AND THE PATH AS WELL TO REACH A SOLUTION

Q1. Consider the following problem: A Water Jug Problem: 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? APPLY DFS ALGORITHM TO REACH THE SOLUTION.

State Representation: The state is represented as **(x, y)**, where:

- **x** = amount of water in the 4-gallon jug.
- **y** = amount of water in the 3-gallon jug.

Allowed Operations:

- Fill the 4-gallon jug: **(4, y)**
- Fill the 3-gallon jug: **(x, 3)**
- Empty the 4-gallon jug: **(0, y)**
- Empty the 3-gallon jug: **(x, 0)**
- Pour water from the 4-gallon jug into the 3-gallon jug until the latter is full or the former is empty.
- Pour water from the 3-gallon jug into the 4-gallon jug until the latter is full or the former is empty.

Solving using DFS:

Starting from the initial state **(0,0)**:

- 1.Fill the 4-gallon jug** $\rightarrow (4,0)$
- 2.Pour water from 4-gallon to 3-gallon** $\rightarrow (1,3)$
- 3.Empty the 3-gallon jug** $\rightarrow (1,0)$
- 4.Pour water from 4-gallon to 3-gallon again** $\rightarrow (0,1)$
- 5.Fill the 4-gallon jug** $\rightarrow (4,1)$
- 6.Pour water from 4-gallon to 3-gallon** $\rightarrow (2,3)$

Name : ADVAIT GURUNATH CHAVAN

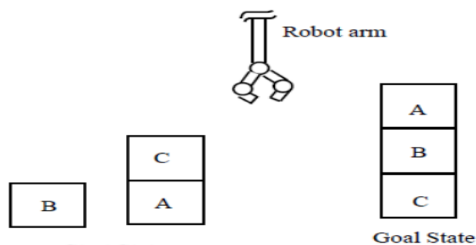
Email ID : advaitchavan135@gmail.com

Path followed to go from **Start State (0,0)** to **Goal State (2,3)**

$(0,0) \rightarrow (4,10) \rightarrow (1,3) \rightarrow (1,0) \rightarrow (0,1) \rightarrow (4,1) \rightarrow (2,3)$

Number of Nodes Expanded $\rightarrow 6$

Q2. Given the blocks world indicated in Fig., it is required to get to goal state from the initial start state. Construct the search tree for: BFS



Assume that following rules for moves will be followed by the robot arm for carrying out this job:

- stack(x, y): stack block x on block y,
- lift(x): lift-up the block x,
- putg(x): put block x on ground,
- unstack(x, y): unstack block x from block y.

•Start State:

- Block **B** is on the ground.
- Block **C** is on **A**, and **A** is on the ground

•Goal State:

- Blocks are stacked as **A on B on C**.

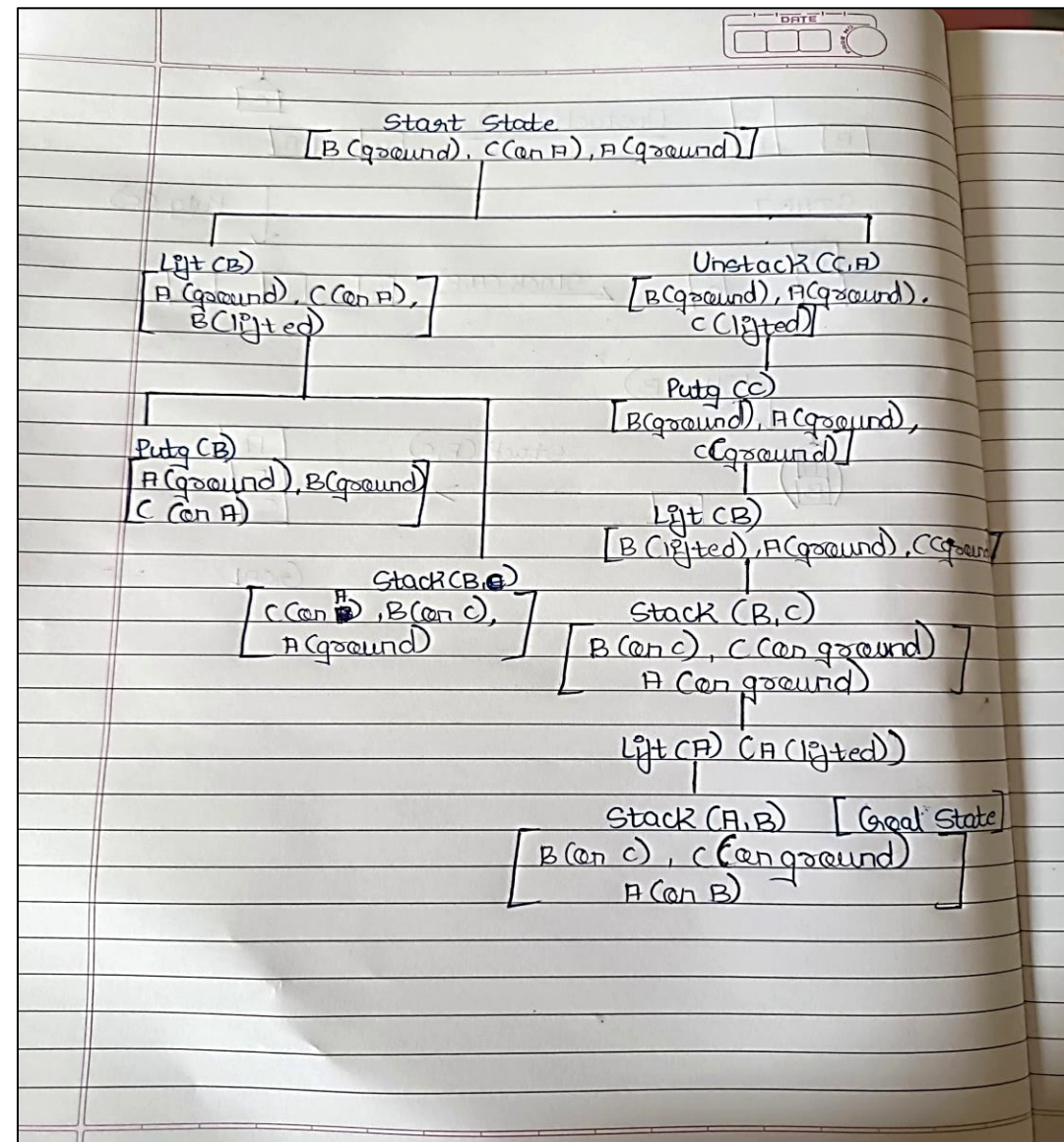
Allowed Operations:

- 1.lift(x) → Lift block x from its current position.
- 2.unstack(x, y) → Unstack block x from block y.
- 3.putg(x) → Put block x on the ground.
- 4.stack(x, y) → Stack block x on block y.

- Unstack(C, A) → C is lifted, now A is free.
- Lift(B) → B is lifted.
- From Unstack(C, A):
 1. Putg(C) → Place C on the ground.
 2. Stack(C, B) → Place C on B.
- From Lift(B):
 1. Putg(B) → Place B back on the ground.
 2. Stack(B, A) → Place B on A.
- Once C is on the ground:
 1. Lift(B) → Stack(B, C)
 2. Lift(A) → Stack(A, B)

Name : ADVAIT GURUNATH CHAVAN

Email ID : advaitchavan135@gmail.com



Q3. Solve using ITERATIVE DEEPENING ALGORITHM

2	8	3
1	6	4
7		5

Initial State

1	2	3
8		4
7	6	5

Goal State

Name : ADVAIT GURUNATH CHAVAN

Email ID : advaitchavan135@gmail.com

DATE: / /

2	8	3
1	6	4
7		5

Start State

↓

(A) Up

2	8	3
1		4
7	6	5

→

2		3
1	8	4
7	6	5

→

	2	3
1	8	4
7	6	5

↓

(D) Down

1	2	3
	8	4
7	6	5

→

1	2	3
8		4
7	6	5

GOAL

Moves Allowed

- ① Up
- ② Down
- ③ Right
- ④ Left

(B) Up

(C) Left

(E) Right

No. of Nodes of Iterative Deepening Search

$$N = (d+1) \times b^d$$

$b \Rightarrow$ Branching factor [Avg. No. of moves per state [3 or 4] ≈ 3.5]

$d \Rightarrow$ depth of solution

In our case, we reach goal state after 5th level of tree

$$\therefore N = (5+1) \times (3.5)^5$$

$$= 6 \times 3.5^5$$

$$= 3151$$

Q4. SOLVE USING BFS

The missionaries and cannibals problem is usually stated as follows. Three missionaries

and three cannibals are on one side of a river, along with a boat that can hold one or

two people. Find a way to get everyone to the other side without ever leaving a group of missionaries in one place outnumbered by the cannibals in that place.

State Representation

Each state is represented as (M, C, B), where:

- M = Number of missionaries on the left side.
- C = Number of cannibals on the left side.
- B = Boat position (1 for the left side, 0 for the right side).

The goal state is (0,0,0), meaning all missionaries and cannibals have crossed safely.

Valid Moves

Possible boat moves (M = Missionary, C = Cannibal):

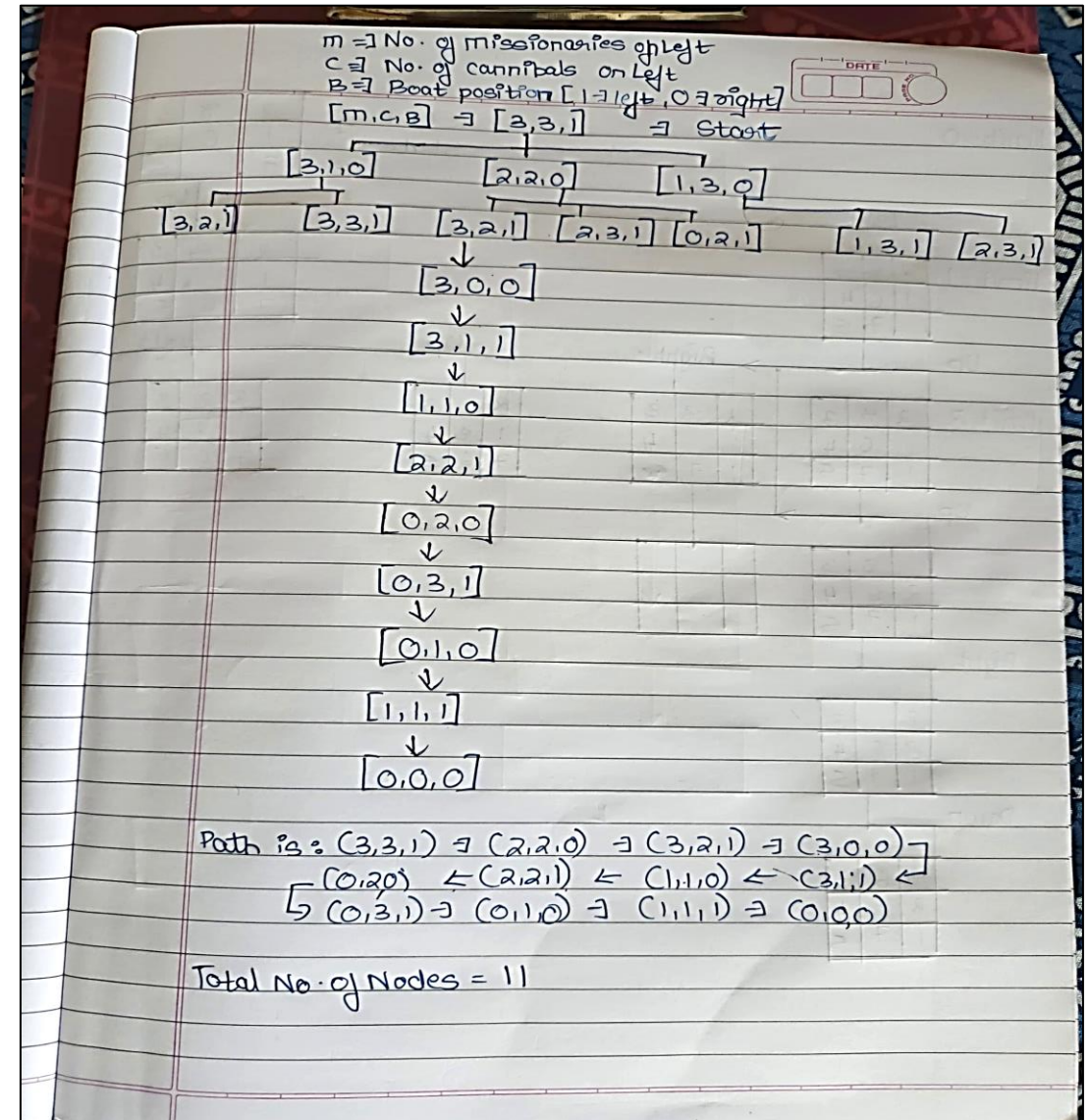
- 1.(1M crosses)
- 2.(2M cross)
- 3.(1C crosses)
- 4.(2C cross)
- 5.(1M, 1C cross)

The move is valid if:

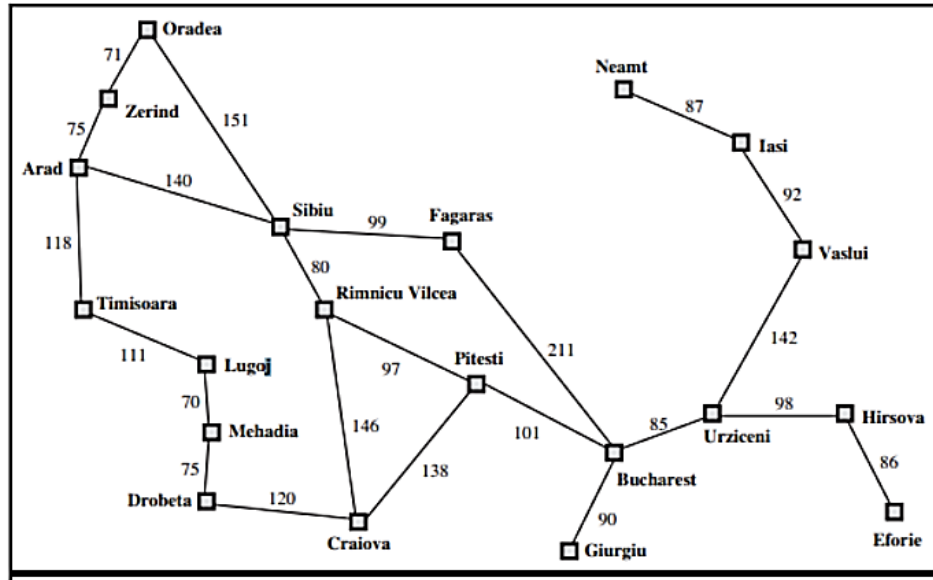
- Missionaries are never outnumbered by cannibals on either side.
- The boat carries at most 2 people.

Name : ADVAIT GURUNATH CHAVAN

Email ID : advaitchavan135@gmail.com



Q5. The map of Romania is given. Start node is Arad and Goal node is Bucharest. Find the optimal path using uninformed search algorithm.



Name : ADVAIT GURUNATH CHAVAN

Email ID : advaitchavan135@gmail.com

Let Arad [Cost = 0] Start-Node
Bucharest End-Node

(A) Arad \rightarrow Sibiu \rightarrow Fagaras \rightarrow Bucharest
 $0 + 140 + 99 + 211 = 450$

(B) Arad \rightarrow Sibiu \rightarrow Rimnicu Vilcea \rightarrow Pitesti \rightarrow Bucharest
 $0 + 140 + 80 + 97 + 101 = 418$

(C) Arad \rightarrow Timisoara \rightarrow Lugoj \rightarrow Mehadia \rightarrow Drobeta \rightarrow Bucharest \leftarrow Pitesti \leftarrow Craiova
 $0 + 118 + 111 + 70 + 75 + 120 + 138 + 101 = 733$

(D) Arad \rightarrow Timisoara \rightarrow Lugoj \rightarrow Mehadia \rightarrow Drobeta \rightarrow Bucharest \leftarrow Pitesti \leftarrow Rimnicu Vilcea \leftarrow Craiova

$0 + 118 + 111 + 70 + 75 + 120 + 146 + 97 + 101 = 838$

Uninformed Search Algorithm uses Uniform Cost Search to find the optimal path from Start-Node to Goal.

So, the path
Arad \rightarrow Sibiu \rightarrow Rimnicu Vilcea \rightarrow Pitesti \rightarrow Bucharest
and its cost is 418.