Tutorial 2:

To understand state space problem formulation

Aim:- To understand state space based problem formulation of AI problems so that problem solving Agent can be applied.

Theory:- First we understand the problem solving agent. Algorithm shown in fig. shows agent program for problem solving agent. Agent first formulates goal and problem, then determines or rather searches an action sequence, after which it returns the next actions to be executed in a sequential maner.

```
function SIMPLE - PROBLEM - SOLVING - AGENT
                    (percept) returns an action
  static: seq. an action sequence, initially empty
          state, some description of the current
                                   world state
          goal, a goal, initially null
          problem, a problem formulation.
  state ←── UPDATE - STATE (state, perceed)
  if seq is empty then do
    goal ←── FORMULATE - GOAL (state)
    problem ←── FORMULATE - PROBLEM (state, goal)


  action ←── FIRST (seq)
  seq ←── REST (seq)
  return action
```

fig. Problem solving Agent Architecture

Defining the problem is referred to as problem formulation. It involves defining following five things:

Initial state :- It is the starting state that the problem is in.

Actions:- It defines all possible actions available to the agent, given it is in some state s currently. It is a function Actions(s) that returns list of all possible actions.

Transition model :- also known as successor function which define which state/s the system tend to move to when a particular action is executed by the agent.

Goal Test :- This act as stopping condition. condition when the state passed to this function is goal state it will return true and sequence under consideration is optimal.

Path Cost :- It is accumulated cost of performing certain sequence of actions. This can help in determining weather the action sequence under consideration is optional.

Thus a problem can formally specified by identifying initial state, actions, transition model

goal test and path cost. Process of finding a solution is called search.

* **working :-** Based on understanding of problem formulation students need to formulate following problems. This will clearly show state space up to depth level 3 as full goal node which ever is shallowest.

1) **8- puzzle problem.**

The problem can be formulated as :-

- **states :** states can be represented by a 3×3 matrix data structure with blank by an underscore :-

1. **Initial state :** { {1, 2, 3}, {4, 8, _}, {7, 6, 5} }

2. **Actions :** The blank space moves in left, right, up & down direction specifying the actions.

3. **successor function :-** If we apply down operation to the start state, the next state has `5` and `_` switched.

4. **Goal test** = { { {1, 2, 3}, {4, 5, 6}, {7, 8, _} }

5. **path cost :** No. of steps to reach to the final state.

- **solution :-**

{ {1, 2, 3}, {4, 8, _}, {7, 6, 5} } →

{ {1, 2, 3}, {4, 8, 5}, {7, 6, _} }

$\{\{1,2,3\}, \{4,8,5\}, \{7,-,6\}\} \rightarrow \{\{1,2,3\}, \{4,-,5\}$
$\{7,8,6\}\}$

$\{\{1,2,3\}, \{4,5,-\}, \{7,8,6\}\} \rightarrow \{\{1,2,3\}, \{4,5,6\},$
$\{7,8,-\}\}$

Path cost = 5 steps.

* 8 puzzle problem :-

Initial state.

Domain

left

Goal State

ii) Navigate to KGCE workshop from HoD IT cabin with minimum number of moves. Can be climbing or alighting, for turning left, right making through a corridor.

state: It can be represented as a top view of the agent along with arrows in directions left, right, forward & backwards. we use 'climb' and 'alight' for moving through stair cases.

1 Initial state :-

Exit ← ———— → Corridor

HoD IT
Cabin
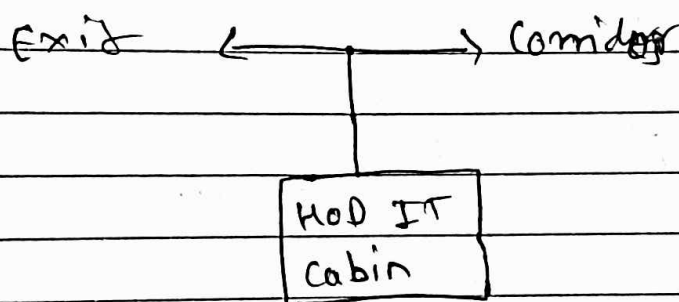
2. Actions : The agent moves in left, right, forward and backward directions along with alighting and climbing the stairs.

3. successor function :- If we apply 'right' operation to the start state, the agent enters the corridor - the first step forwards goal state.

4. Goal test

workshop

↓

Corridor
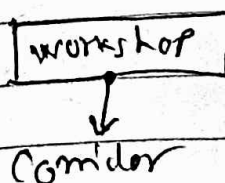
5. path cost : No. of actions to reach the workshop

path cost = 8 direction + 4 staircase

= 12

state space :-

HOD 2r Cabin — Initial state

Left

Exit

Right

Corridor

Left + Alight   Right   forward   Backward

Lab A   Corridor   Exit

Lam

Left   Backward   Right + climb

Exit   Corridor   stage 1

workshop

Goal state

HOD IT cabin → KGCE workshop (obviously)

① Corridor

Exit ← ———— >>> → Corridor

┌──────┐
│HOD IT│
│Cabin │
└──────┘

Lawn ←←← ┌────────┐ → Labs
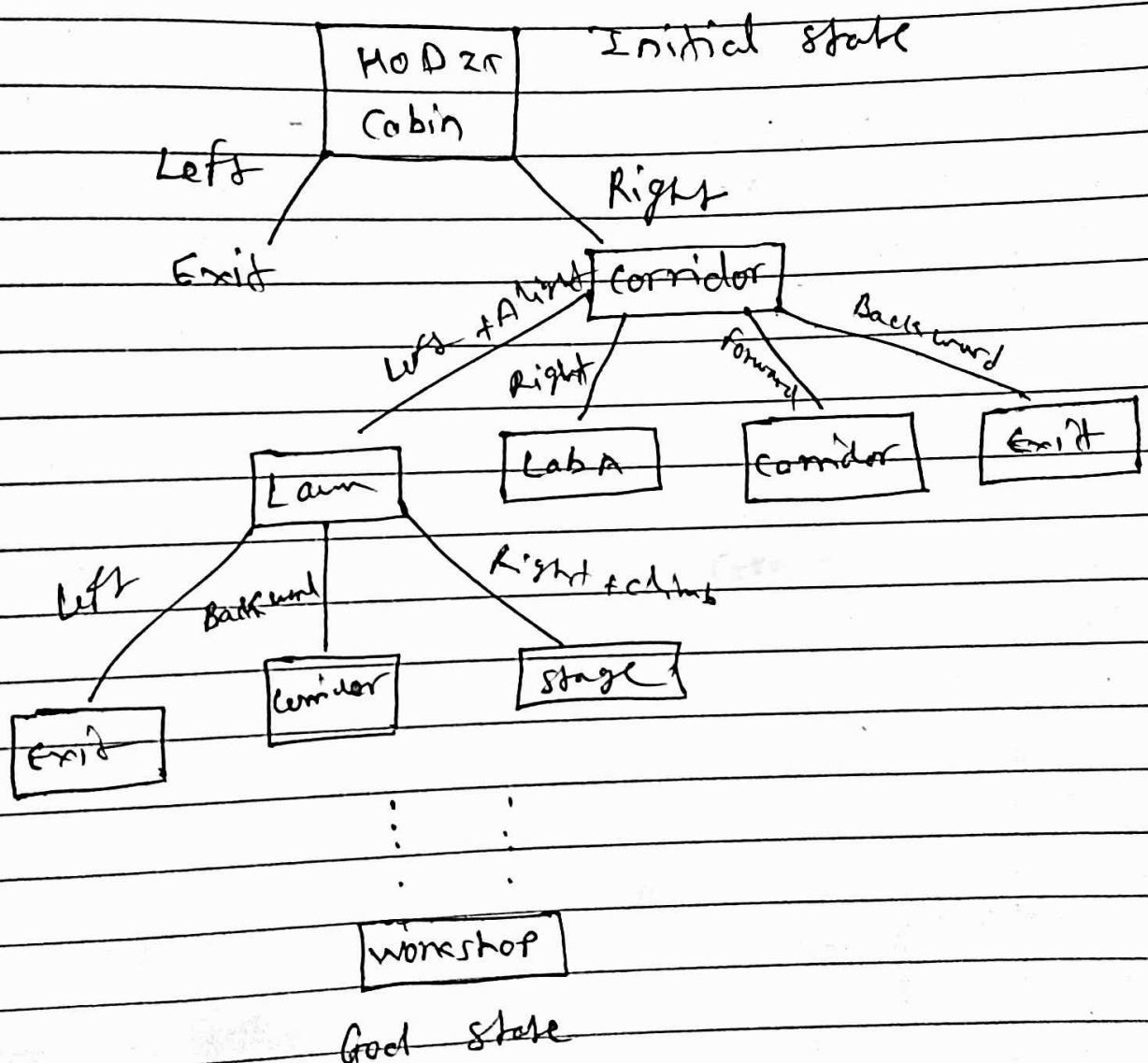   ALign │Corridor│
           └────────┘
              ↓
            Exit

② ③ Corridor

Exit ← ┌─────┐ >>>>> stage   ↑
       │Lawn │   ┌─────┐
       └─────┘   CLIMB   Compo←←← │Stage│
          ↓   DEPT ALIGN└─────┘
       Corridor   ↓
                                      Lawn

④ ⑤

EP LAB
↑
┌──────┐   Corridor←←← ┌─────┐
│Comps │ >>>> Alley   ←← │Alley│
│Dept │   ALIGN   └─────┘
└──────┘   ↓
Corridor ←   Comps Dept
   ↓
 Stage

⑥ Corridor →>>>>> Corridor   ⑦   workshop
                                        ↑
┌────────┐   ↑
│Corridor│   ┌────────┐
└────────┘   │Corridor│→ Hthur
    ↓   └────────┘
  Alley

⑧ ┌────────┐   Goal state.
   │workshop│
   └────────┘
       ↓
    Corridor