Name:- Divyansh Tiwari
Roll No:- 2020111002

MDL

# Assignment - 2

(Note:- Page Nos. are written on top of each page.)

Q.1) Express . . . . . path cost).

⇒ • States:-

1. Holding(A) : the block A is held in robot's arm.
2. Empty(@) : the robot's arm is empty.
3. OnTable(A) : the block A is kept on the table.
4. OnBlock(A,B) : the block A is on block B.
5. Clear(A) : there is nothing on top of block A.

• Actions:-

1. PickUP(A) : the robot's arm picks up block A.
2. Put Down(A) : block A is put down on the table at a free space.
3. Put On(A,B) : put block A on block B.
4. Remove(A,B) : remove block A from the block B.

• Initial state:- The given configuration is the initial state.

• Goal state:- The goal state is the configuration that is needed to be achieved.

• Path cost :-

Let a unit cost is charged (say, equal to 1) for moving a block from one position to another.

Therefore,

Path cost

$$\Rightarrow \begin{cases} 1 & ; \text{ movement happens} \\ 0 & ; \text{ otherwise} \end{cases}$$

The optimal solution is the case with the minimum path cost required to go from the initial state to the goal state.

(1.2) Given the following ..... 3 iterations each:

⇒ Initial state :- $\begin{bmatrix} C \\ A\ B \end{bmatrix}$ Final state :- $\begin{bmatrix} A \\ B \\ C \end{bmatrix}$

(a) Breadth - first :-

1st - iteration :- $\begin{bmatrix} B & & & & \\ C & C & & C & C \\ A & AB & ABC & AB & AB \end{bmatrix}$

2nd - iteration :- $\begin{bmatrix} & & & B & \\ C & & C & C & C \\ AB & ABC & AB & AB & A & AB \end{bmatrix}$

– P.T.O.

3rd - iteration :-

∵ configuration $\begin{bmatrix} C \\ AB \end{bmatrix}$ is already visited

∴ we skip it & move to the next config.

$$\begin{bmatrix} C & C & C & C & B & B & C & C & A & A \\ AB, & AB, & A, & AB, & AC, & ACB, & AC, & AB, & AB, & ABC, & BC, & BC, & ABC \end{bmatrix}$$

(b.) Depth - first :-

1st - iteration :- $\begin{bmatrix} C & C & C & C \\ A, & AB, & ABC, & AB, & AB \end{bmatrix}$

2nd - iteration :- $\begin{bmatrix} B \\ C & C & C & C & C \\ A, & AB, & AB, & ABC, & AB, & AB \end{bmatrix}$

3rd - iteration :-

∵ Node $\begin{bmatrix} B \\ C \\ A \end{bmatrix}$ is already visited in our DFS

∴ we skip it & move it to next node.

Also, Node $\begin{bmatrix} C \\ AB \end{bmatrix}$ is also visited earlier & this happens twice.

∴ we go on next node which becomes [ABC]

$$\Rightarrow \begin{bmatrix} B & B & C & C & A & A & C & C \\ AC, & ACB, & AC, & AB, & AB, & ABC, & BC, & BC, & ABC, & AB, & AB \end{bmatrix}$$

(c.) Uniform search:-

∵ No. of steps required to reach the config. = cost

⇒

1st iteration:-

| | | | | |
|---|---|---|---|---|
| C | C | C | A | C |
| AB | AB | AB | B | CAB |
| cost: 1 | 0 | 0 | 1 | 1 |

2nd iteration:-

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | C | | C | C | A | C | A | C |
| ABC | AB | AB | AB | B | AB | AB | B | CAB |
| cost: 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |

Nodes which have been visited before have been skipped.

Now, on removing all the nodes which have been visited before:-

| | | |
|---|---|---|
| A | | |
| C | | |
| B | ABC | |
| cost 1 | 1 | |

3rd iteration:-

| | | |
|---|---|---|
| A | | |
| C | C | |
| B | BA | ABC |
| cost: 1 | 1 | 1 |

Q.3) Heuristic 1:- All the blocks which are not in their correct position as per the goal state are counted. Those which are in robot's arm are not counted.

Heuristic 2:- We calculate the distance ~~both~~ between the current state of block & the final state of block while looking at the details of each block.

Example:- initial state:- $\begin{bmatrix} & C \\ A & B \end{bmatrix}$

final state:- $\begin{bmatrix} C \\ A \\ B \end{bmatrix}$

According to heuristic (1),

A & C are at incorrect position

$\Rightarrow$ count = 2.

According to heuristic (2),

According to final state, A is above B & below C & it is seen that B & C are at incorrect position. $\Rightarrow$ count = 2.

For C, no block is at the top $\Rightarrow$ 1 is added.

For B, block above it does not match $\Rightarrow$ 1 is added

Q.4) Initial state:- $\begin{bmatrix} C \\ A\ B \end{bmatrix}$ Final state: $\begin{bmatrix} A \\ B \\ C \end{bmatrix}$

Using heuristic (1.),

1st iteration:-

$$\begin{bmatrix} B \\ C & C & C & C \\ A, AB, ABC, AB, AB \end{bmatrix}$$

cost: 3, 3, 2, 3, 3

Expanding the node with the cost = 2.

2nd iteration:-

$$\begin{bmatrix} B & B & C & C & A & A & B & C & C & C & C \\ AC, ACB, AC, AB, AB, ABC, BC, BC, ABC, A, AB, AB, AB \end{bmatrix}$$

cost: 1, 2, 2, 3, 3, 2, 2, 2, 2, 3, 3, 3, 3

Removing all the visited nodes (expanding with cost = 1)

3rd iteration:-

$$\begin{bmatrix} A & & & & & & & & B \\ B & B & B & B & C & C & A & A & C \\ ACB, AC, C, AC, AC, AB, AB, BC, BC, A \end{bmatrix}$$

cost:- 2, 1, 0, 1, 2, 3, 3, 2, 2, 3