UGC Paper 1st Free Cl...
only admins can send messages

+91 81453 66384 joined using this group's invite link

+91 70102 37343 joined using this group's invite link

+91 96672 47765 joined using this group's invite link

+91 98557 99207 joined using this group's invite link

+91 60035 13791 joined using this group's invite link

+91 83590 38670 joined using this group's invite link

+91 91497 27505 joined using this group's invite link

+91 70910 66218 joined using this group's invite link

+91 75779 16791 joined using this group's invite link

+91 60035 13791 left

+91 90012 26665 joined using this group's invite link

+91 80037 25657 joined using this group's invite link

+91 89555 46730 joined using this group's invite link

Type a message

UGC Paper 1st Free Cl...
120 subscribers

December 28

Channel created

Channel photo changed

Broadcast

government_job_2020

1,711 Posts

6,845 Followers

7 Followi

Govt job 2020 (Fillerform) 17K

Education Website

Free Online Computer Class

1. Baisc computer

2. Web development

3. Hackig ... more

youtu.be/mIfPC5C-EvQ

Jaipur, Rajasthan

Edit Profile

Promotions | Insights | Contact

New

1 5 K Sub

YouTube

2000 users

UGC NET 100% Off Free Class

Free Notes

Live Class

5000+MCQ+PYQ

Free Books

fillerform

HOME WORK

39.	What is the best method to go for the game playing problem?
a.	Optimal Search
b.	Random Search
c.	Heuristic Search
d.	Stratified Search

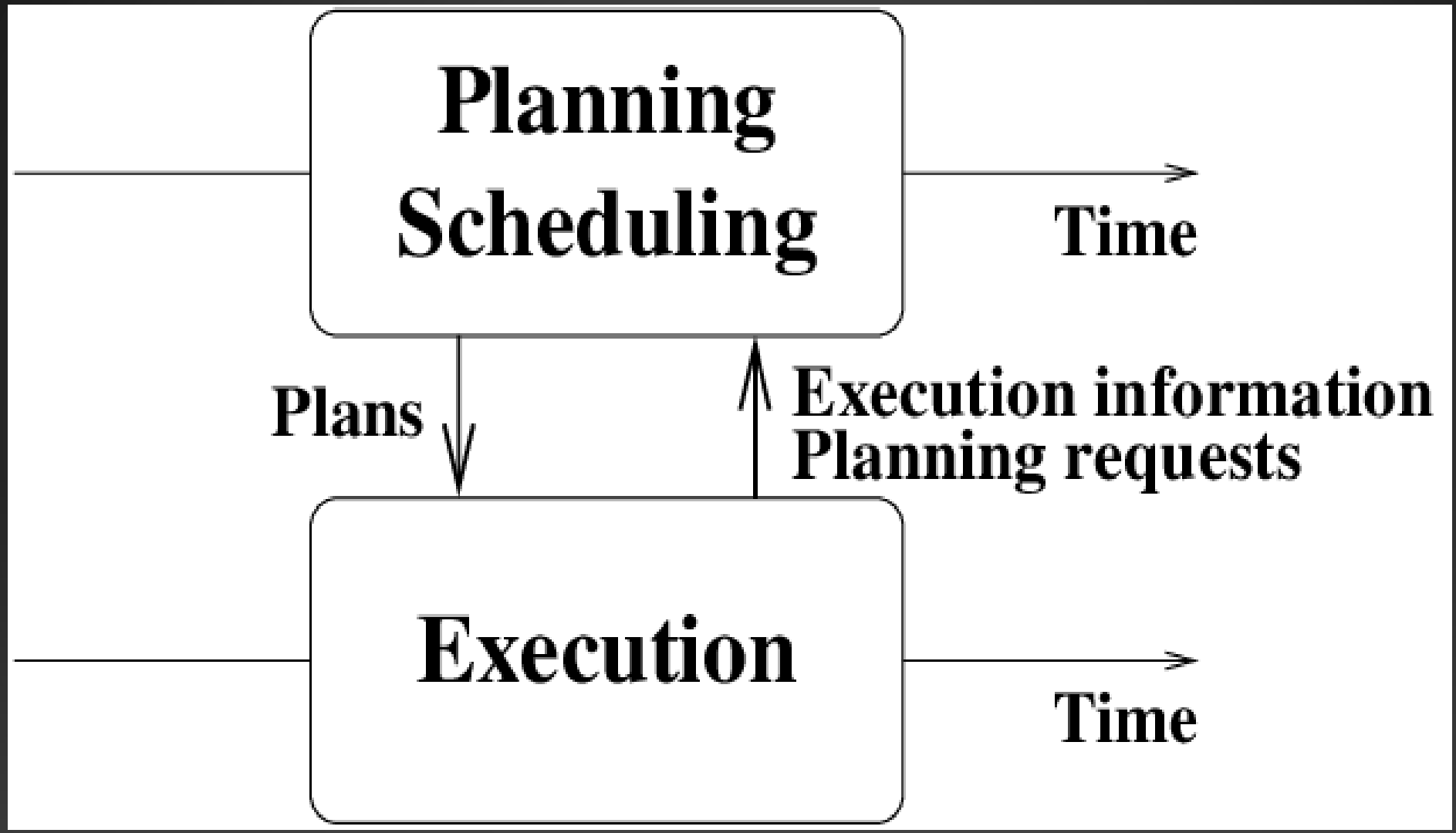
❑ Approaches to AI

Content:

1. Planning in AI

WHAT IS PLANNING IN AI?

- The planning in Artificial Intelligence is about the decision making tasks performed by the robots or computer programs to achieve a specific goal.
- The execution of planning is about choosing a sequence of actions with a high likelihood to complete the specific task.



Components of Planning System

- **Choose the best rule** to apply next based on the best available heuristic information.
- **Apply the chosen rule** to compute the new problem state that arises from its application.
- **Detect** when a solution has been found.
- **Detect dead ends** so that they can be abandoned and the system's effort directed in more fruitful directions.
- **Detect when** an almost correct solution has been found and employ special techniques to make it totally correct.

1.CHOOSING RULES TO APPLY

- In order to select appropriate rules first isolate a set of difference between the desired goal state and current state and then identify those rules that are relevant to reducing those differences.

- If several rules are found, a variety of other heuristic information can be exploited to choose among them.

2. APPLYING RULES

- In simple system, applying rule is easy. Each rule simply specified the Problem state that would result from its applications.
- In complex system, we must be able to deal with rules that specify only a small part of the complete problem state.
- One way is to describe, for each action, each of the changes it makes to the state description.

DETECT A SOLUTION

- Find a solution to a problem when is has found a sequence of operators that transforms the initial problem state into the goal state.
- One of the representatives systems for planning system is, predicate logic.

DETECT DEAD ENDS

- The exploring path that can never lead to a solution.
- No indication of goal Node.
- If the search process is reasoning forward from the initial state, it can prune any paths that leads to a state from which the goal state cannot be reached.
- If the search process is reasoning backwards from the goal state it can also terminate a path either because it is sure that the initial state cannot be reached.

REPAIRING AN ALMOST CORRECT SOLUTION:

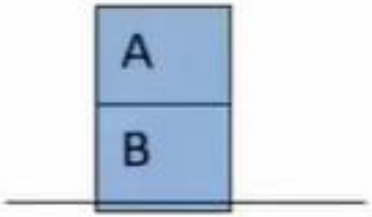
- A slightly better approach is to compared the desired solution and derived solution and if there is a difference then the problem solving system can be called again and asked to find a way of eliminating this new difference. The first solution could be combined with the second one to form a solution to the original problem.

BLOCKS-WORLD PLANNING PROBLEM

- The blocks-world problem is known as **Sussman Anomaly.**
- **There is a flat surface on which blocks can be placed.**
- **There are a number of sequare blocks, all the same size.**
- **They can be stacked one upon the other.**
- **There is robot arm that can manipulate the blocks.**

Actions of the robot arm

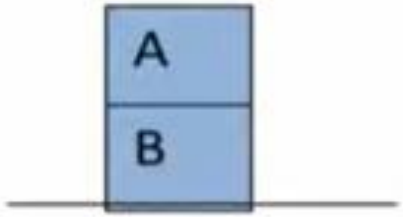
- UNSTACK(A,B)
- STACK(A,B)
- PICKUP(A)
- PUTDOWN(A)



- Notice that the robot arm can hold only one block at a time.

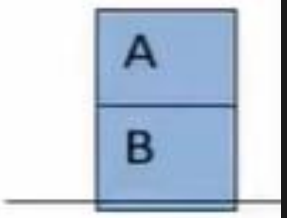
Predicate

- In order to specify both the conditions under which an operation may be performed and the results of performing it, we need the following predicates:
- ON(A,B)
- ONTABLES(B)
- CLEAR(A)
- HOLDING(A)
- ARMEMPTY



- ON(A,B) ^ ONTABLE(B)^CLEAR(A)

If we execute UNSTACK(A,B) in this state



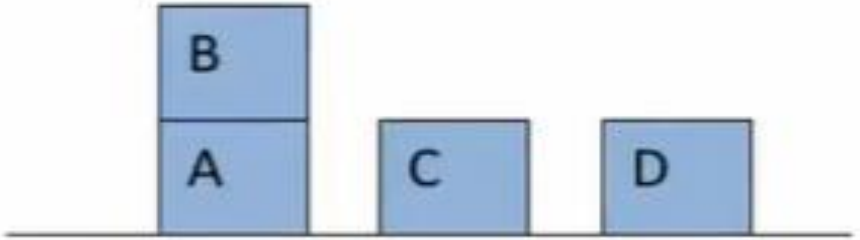
Then,

HOLDING(A)^CLEAR(B)

Goal Stack Planning

- To start with goal stack is simply:

ON(C,A)^ON(B,D)^ONTABLE(A)^ONTABLE(D)



ONTABLE(A) and ONTABLE(D) are already true in the initial state.

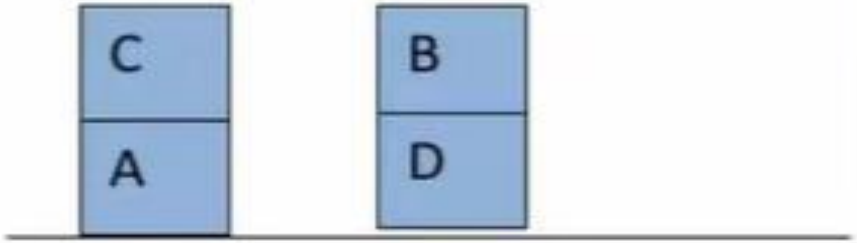
- **Alternative 1: Goal Stack:**

ON(C,A)

ON(B,D)

ON(C,A)^ON(B,D)^OTAD

- **Alternative 2: Goal stack:**



ON(B,D)

ON(C,A)

ON(C,A)^ON(B,D)^OTAD

Goal Stack Planning

- Next we see if CLEAR(A) is true. It is not. The only operator that could make it true is UNSTACK(B,A). This produces the goal stack:

ON(B,A)

CLEAR(B)

ON(B,A)^CLEAR(B)^ARMEMPTY

UNSTACK(B,A)

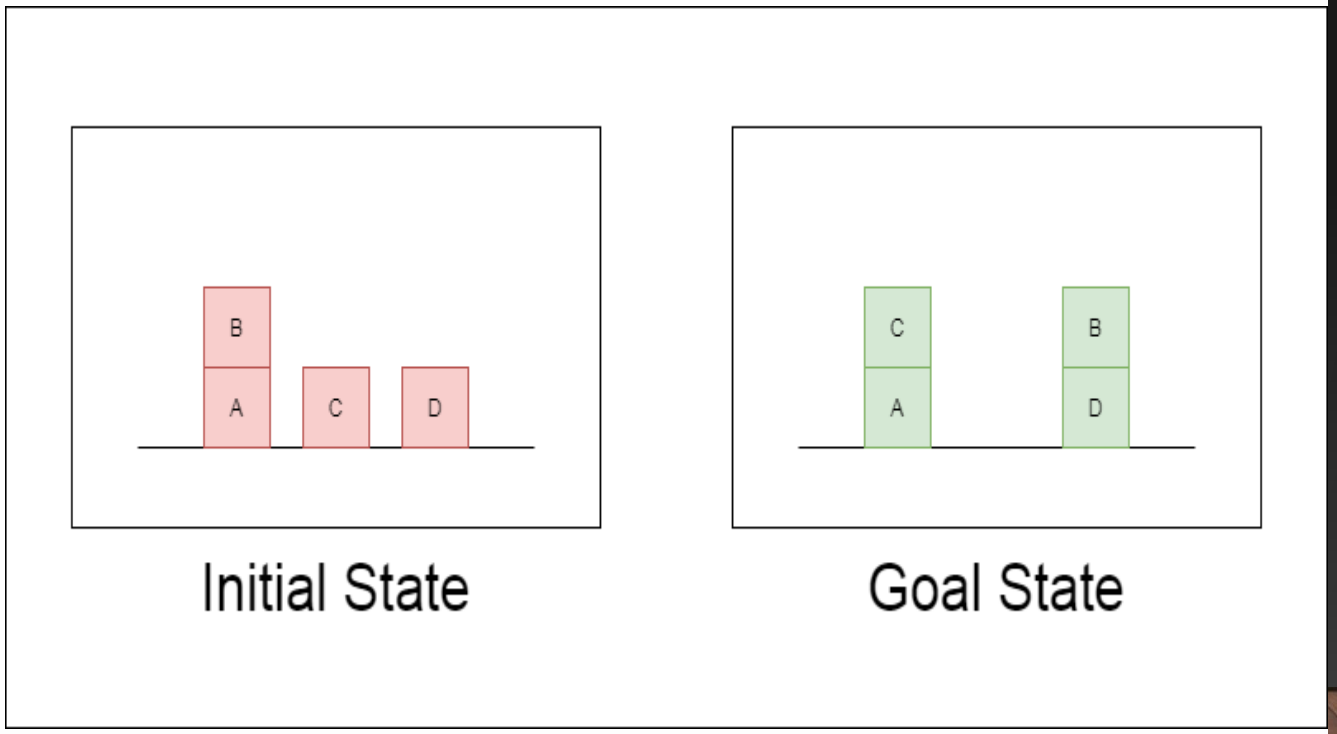
HOLDING(C)

CLEAR(A)^HOLDING(C)

STACK(C,A)

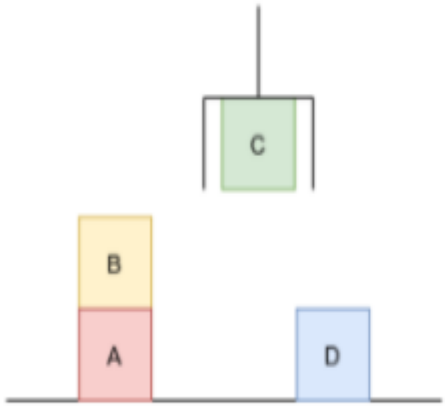
ON(B,D)

ON(C,A)^ON(B,D)^OTAD



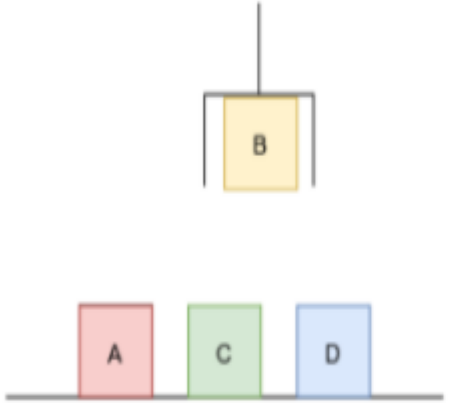
Initial State

Goal State

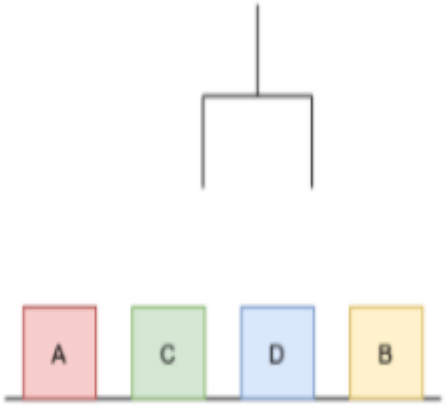


1 - PICKUP(C)

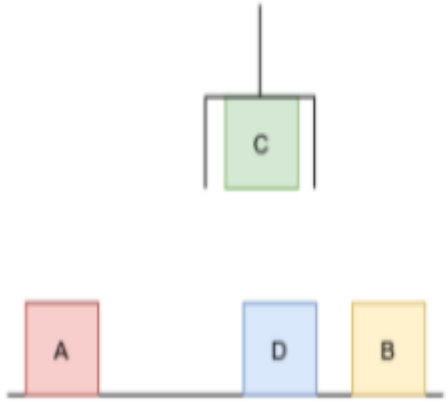
2 - PUTDOWN(C)



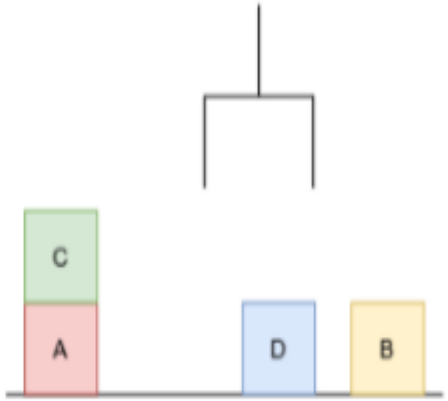
3 - UNSTACK(B,A)



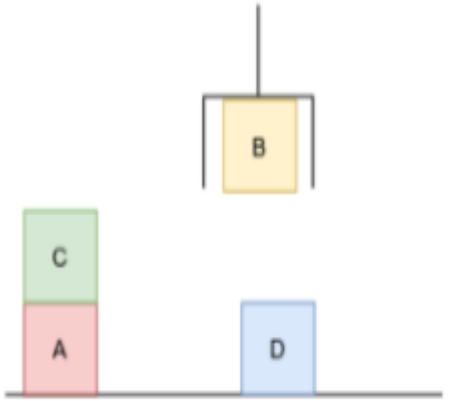
PUTDOWN(B)



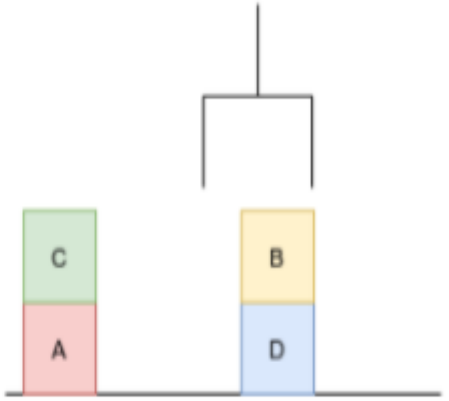
5 - PICKUP(C)



6 - STACK(C,A)



7 - PICKUP(B)



8 - STACK(B,D)

Goal Stack Planning

ALT1:	ALT2:
ONTABLE(C)	ON(C,x)
CLEAR(C)	CLEAR(C)
ARMEMPTY	ARMEMPTY
ONTABLE(C) ^CLEAR(C)^AR MEMPTY	ON(C,x)^CLEA R(C)^ARMEMP TY
PICKUP(C)	UNSTACK(C,x)
CLEAR(A)^HOL DING(C)	CLEAR(A)^HOL DING(C)
STACK(C,A)	STACK(C,A)
ON(B,D)	ON(B,D)
ON(C,A)^ON(B,D)^OTAD	ON(C,A)^ON(B, D)^OTAD

HOME WORK

2.	An expert system shell is an expert system without
a.	domain knowledge
b.	explanation facility
c.	reasoning with knowledge
d.	all of the above


FEEDBACK

For More Information

www.ugc-net.com

 /Fillerform  /Fillerform /Fillerform

 info@fillerform.com

 8209837844