

FORWARD CHAINING / REASONING

→ It is a form of reasoning which starts with atomic sentences in the knowledge base & applies inference rule in the forward direction to extract more data until a goal is reached.

* It comes with the known facts and iterates the process to find new facts until it reaches its goal.

* Popular for expert system, business & production rule systems.

* Properties:

- 1) It moves from bottom to top.
- 2) It is a process of making a conclusion based on a known fact of data by starting from initial state & reach the goal state.
- 3) Forward chaining approach is also called as data-driven as we reach the goal using available data.
- 4) used in expert system.

Examples:

Rule 1: If A & C then F

2: If A & E then G

3: If B then E

4: If G then D

Database

AB

KB

A & C → F

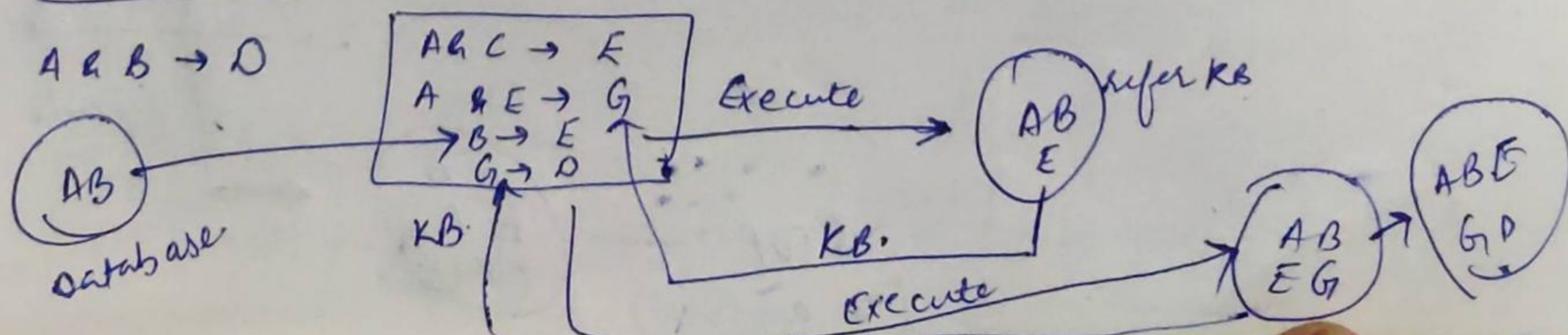
A & E → G

B → E

G → D

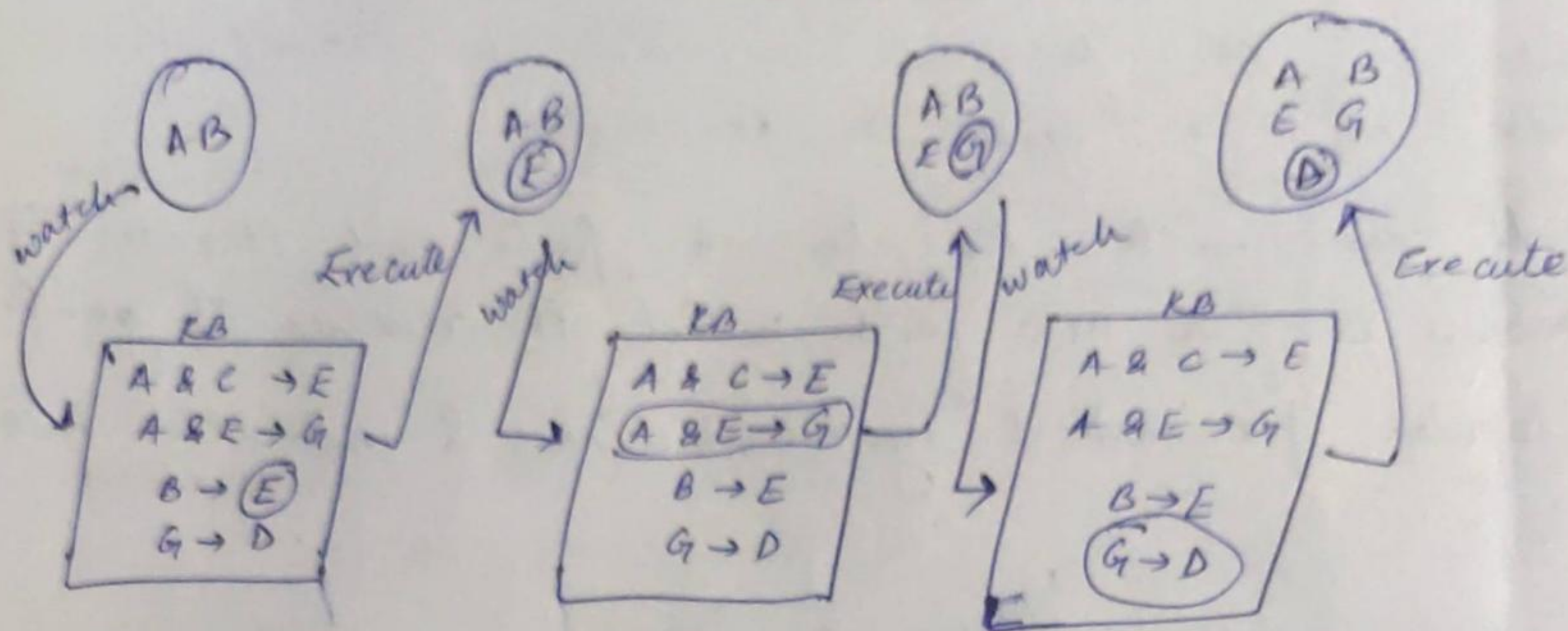
Knowledge Base

* Problem: Prove A & B true, then D is true.



Forward Chaining Format:

① Problem solved using this finally D is true as A & B were true.



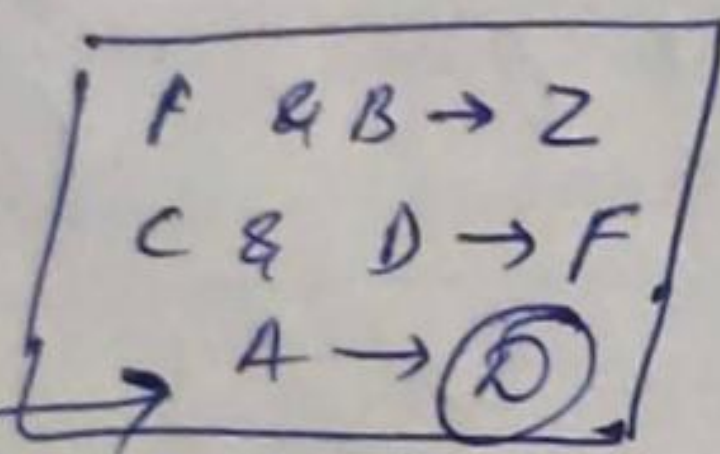
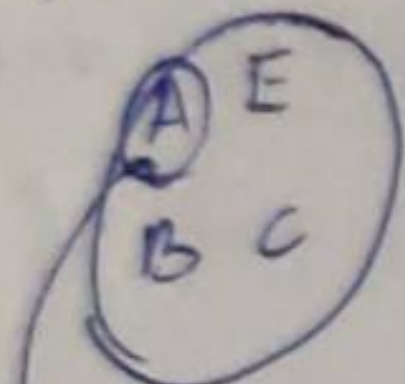
② Goal state = Z

Termination Condition : Stop if Z is derived.

(No further rule can be applied).

Fact

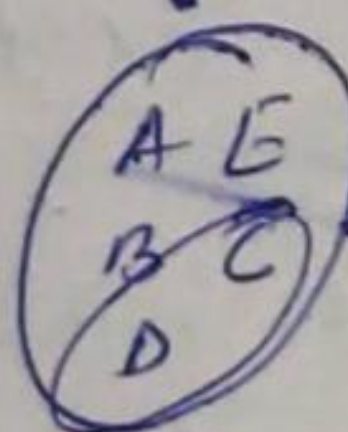
Rules:



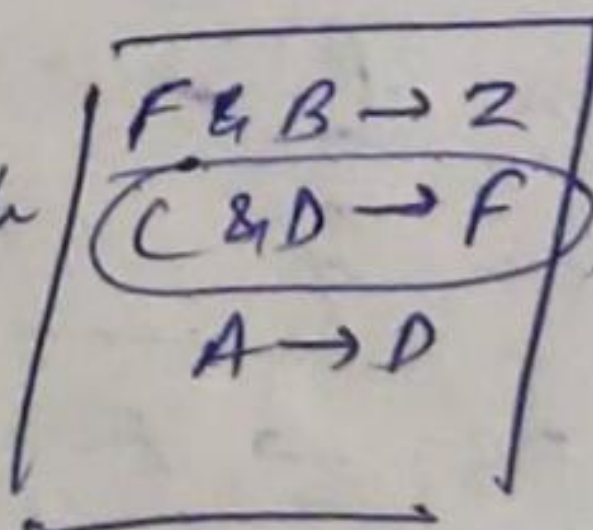
= Given

watch

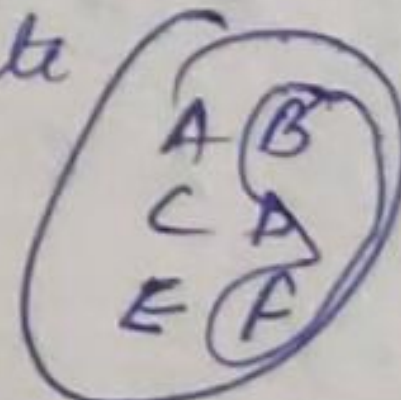
execute



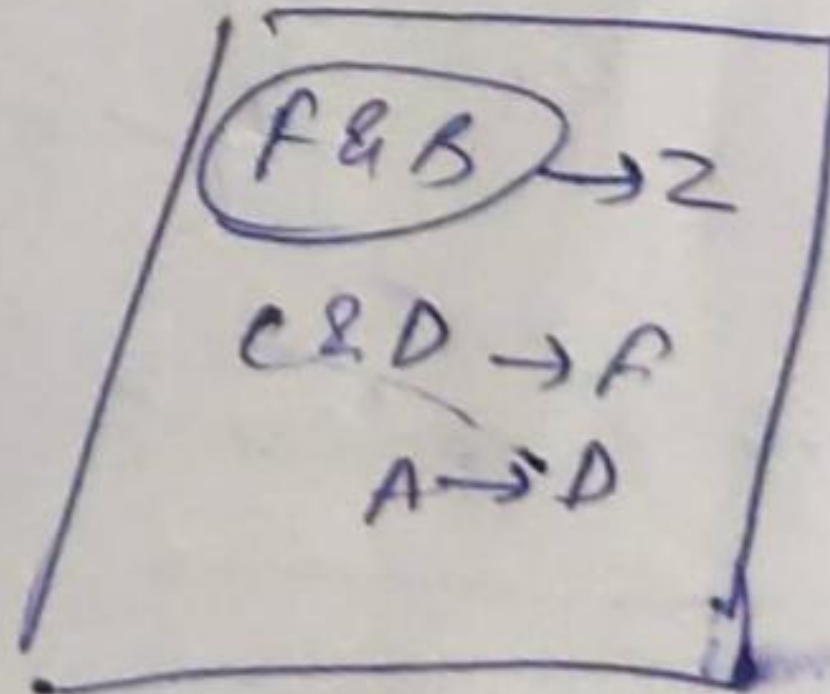
watch



execute

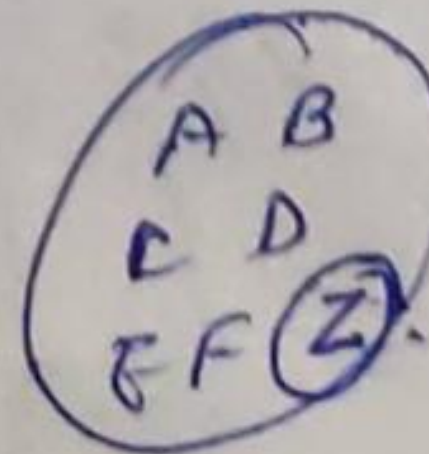


watch



execute

STOP — (Z arrived)



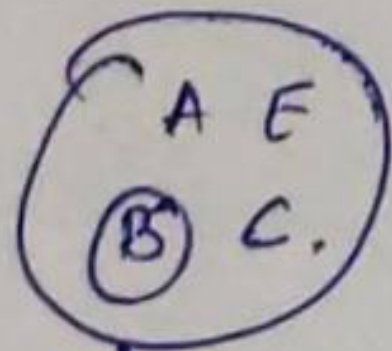
BACKWARD CHAINING.

- * It is a form of reasoning which starts with the goal and works backward; chaining through rules to find known fact support the goal.
- * It is implemented in logic programming by SLD resolution. Both rules are based on modus ponens inference rule.
- * It is

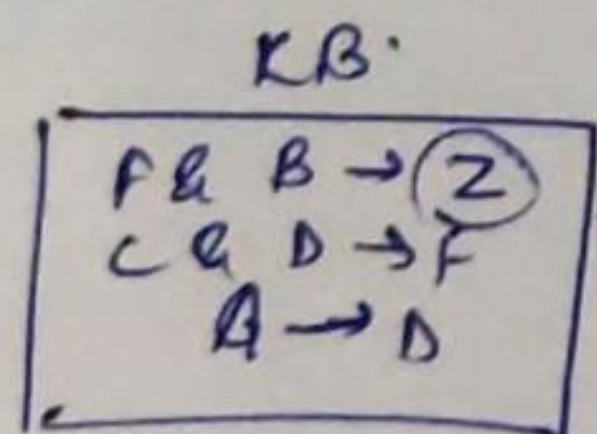
* Properties of Backward Chaining:

- 1) It is known as Top-Down Approach.
- 2) Based on Modus Ponens inference rule.
- 3) In this, the goal is broken into sub-goal or sub-goals to prove the facts true.
- 4) Called as Goal-Driven, as a list of goal decides which rules are selected & used.
- 5) It is used in Game Theory, Automated Theorem Proving tools, Inference Engines, Proof Assistant, etc.
- 6) It mostly used a depth first search strategy for proof.

Example: Fact

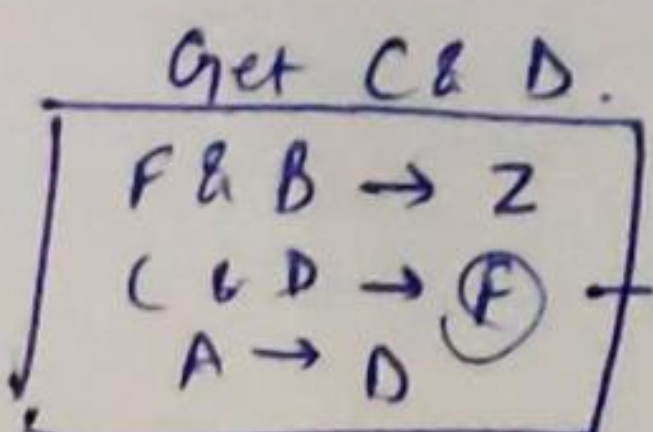


Goal state: Z.

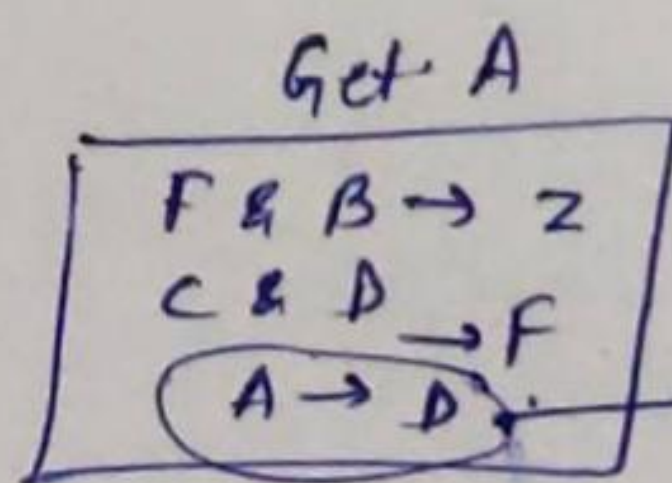
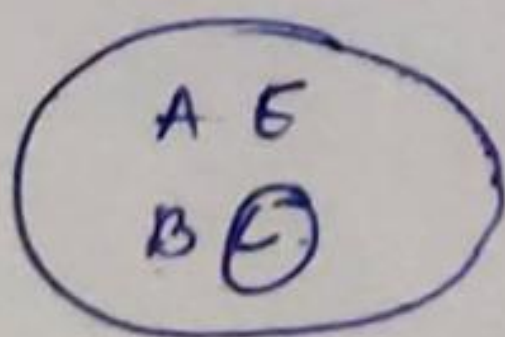


want Z.

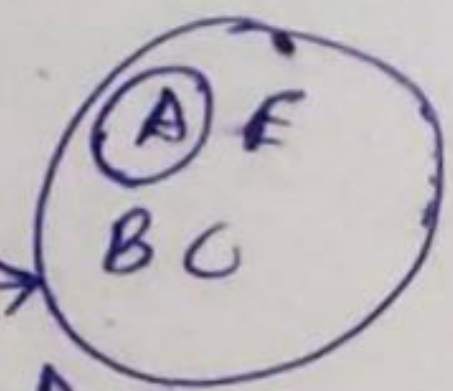
Get F & B.



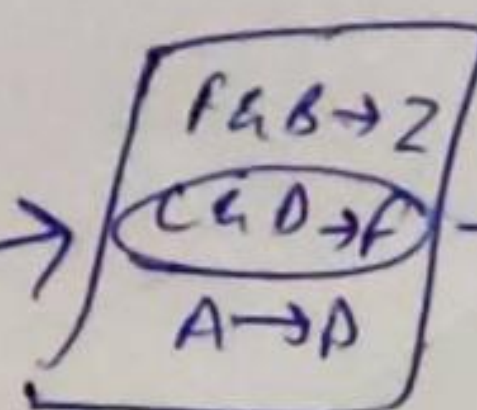
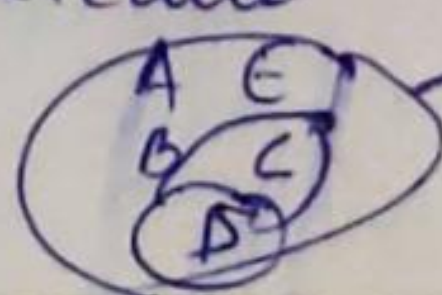
want F.



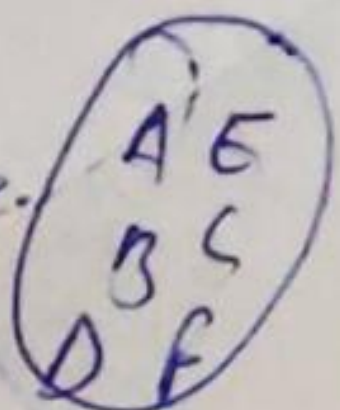
want D.

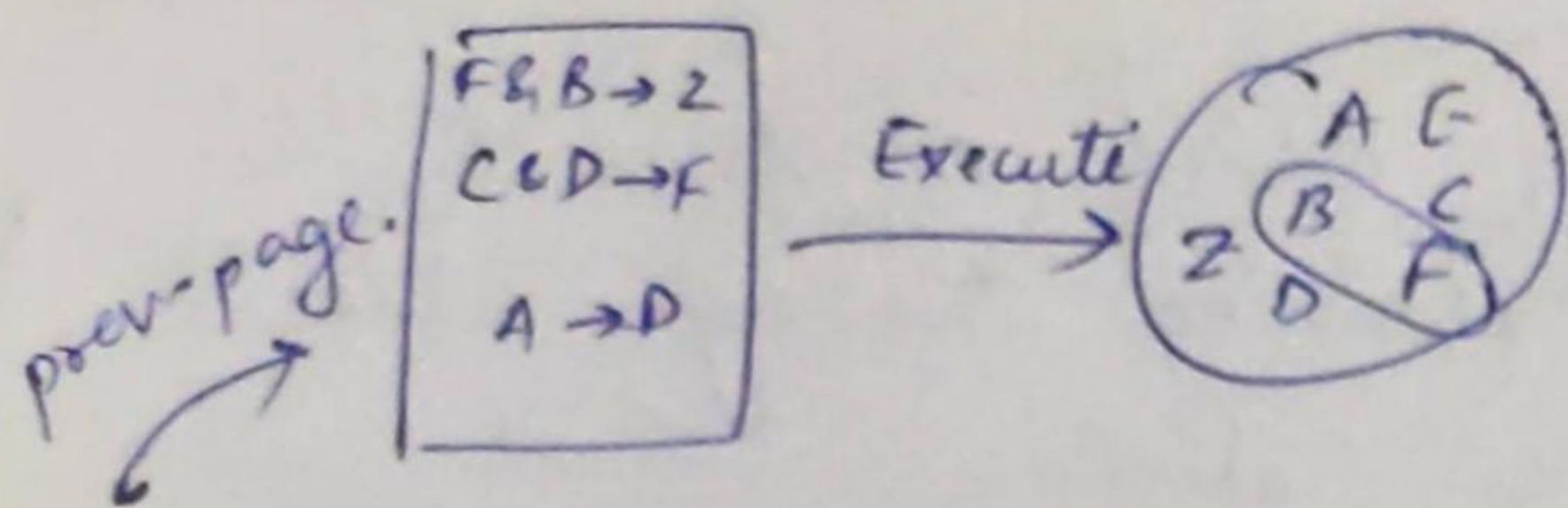


Execute



Exe.





- 1) focus on what element you want to get; focus on goal and get its elements from facts.
 - 2) B found in fact; but F not present so next target'll be to get F from Knowledge Base (KB).
 - 3) from 2nd rule; $C \& D \rightarrow F$, we'll select C & D. and from fact we'll get C, but D is not present here. so next target will be D.
 - 4) From 3rd rule, $A \rightarrow D$; we'll select A & get it from fact. Thus, A is present there.
-