Shubham Shankar

10017610 68

Assignment

Task 1

Part a)

	KB	S_i	
①	T	T	✓
②	F	T	
③	T	T	✓
④	F	T	
⑤	F	F	
⑥	F	F	
⑦	T	T	✓
⑧	F	F	

$KB \models S_i$, KB entails S_i, when ever KB is true & S_i is also true.

In the above ~~also~~ Table, for ①, ③, ⑦, (S_i is true when KB is True.

Thus we can say that KB entails S_i

Part b) Not (KB) will entail Not (S_i) if & only if Not(S_i) is true, for when Not (KB) is true

	Not (KB)	Not (Sᵢ)	
1)	F	F	
2)	T	F	X
3)	F	F	
4)	T	F	X
5)	T	T	
6)	T	T	
7)	F	F	
8)	T	T	

From the above table, we see that there are 5 places where Not (KB) is true, But Not(Sᵢ) is not true in all those 5 places.

This means that Not(KB) does not entail Not(Sᵢ)

∴ Summary,

KB entails S_1

Not (KB) does not entail S_1

② **Task 2**

There are 2 cases where KB is false

Case 1) : A = true, B = false, C = false, D = true
Case 2) : A = false, B = false, C = true, D = false.
In all other cases KB is true.

For the given KB, Let X be the CNF

$$X = ((A \cap \neg B \cap \neg C \cap D) \cup (\neg A \cap \neg B \cap C \cap \neg D))'$$

$$X = ((A \cap \neg B \cap \neg C \cap D)' \cap (\neg A \cap \neg B \cap C \cap \neg D)')$$

$$X = ((\neg A \cup B \cup C \cup \neg D) \cap (A \cup B \cup \neg C \cup D))$$

③ **Task 3**

A $\iff$ B
B $\Rightarrow$ C
D $\Rightarrow$ A
C And E $\Rightarrow$ F
E
D

i) Forward chaining

Step 1
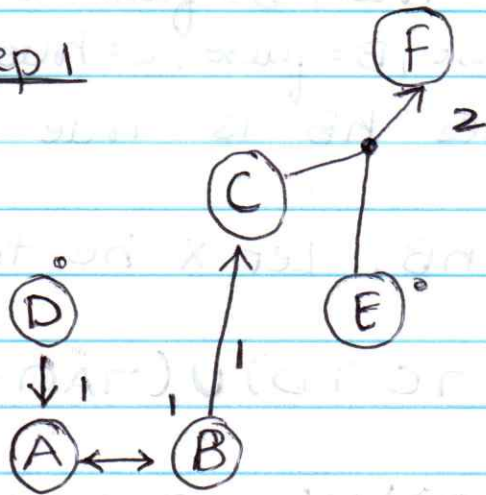
Initially Arc D & E have Value zero.
Then in next step Reduce the Arc value
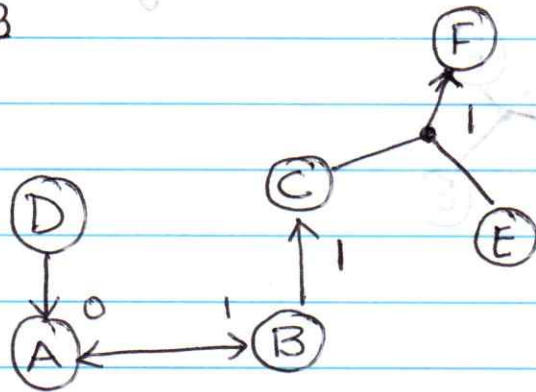which has E as a Starting point.

Step 2
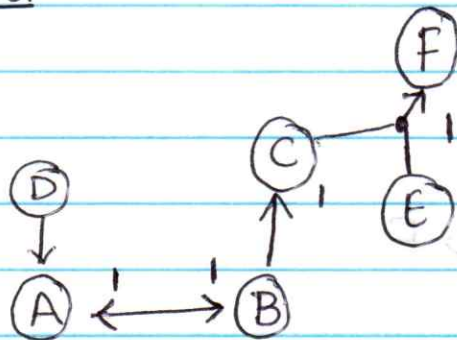
Now here we see D has Value zero.
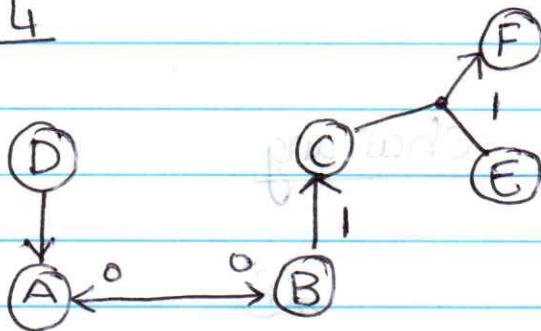So Reduce the Value of Arch which
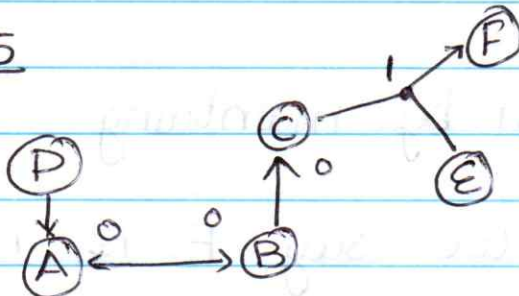
has D as starting point.

Step 3

here A has Value zero, reduce the Value
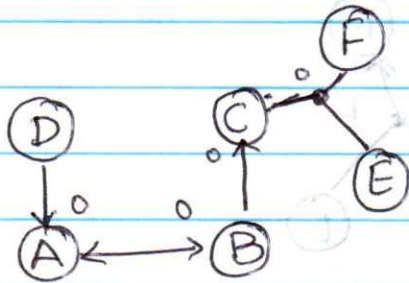of Arc which has A as starting point.

Step 4

here B has Value zero, reduce the Value of
Arc with B as starting point

Step 5

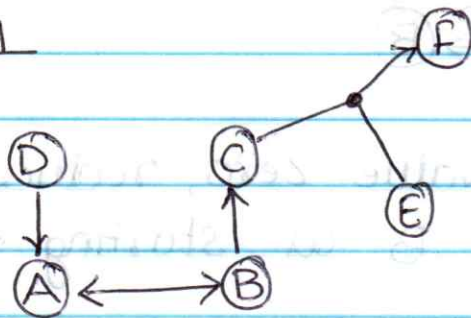here C has value zero, now reduce the Arc, with C as a starting point

Finally F becomes zero,

Thuse we can say that $\boxed{\text{KB entails F}}$
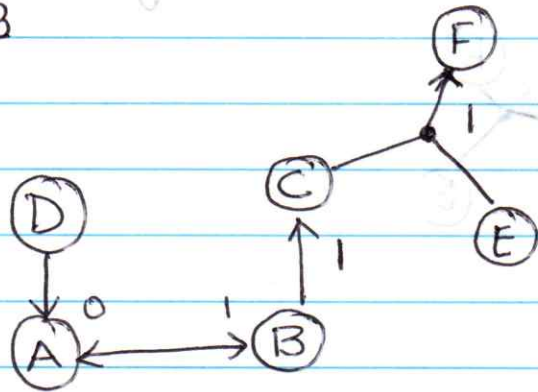
ii) **Backward Chaining**

Step 1

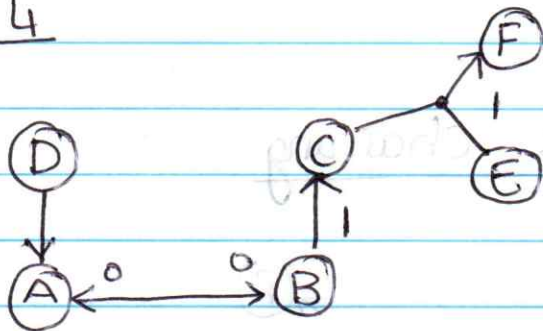we start by maintaing goal stack.

Initially we say F is true
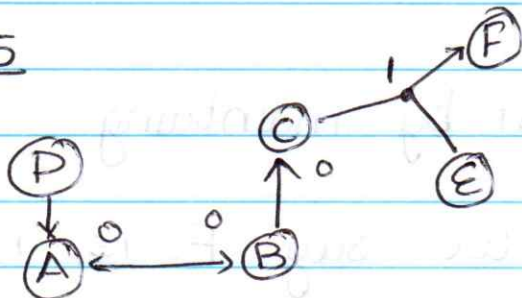
has D as starting point.

Step 3

here A has Value zero, reduce the Value of Arc which has A as Starting point.
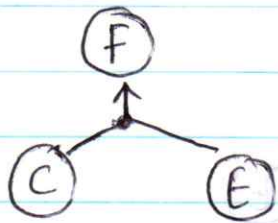
Step 4

here B has Value zero, reduce the Value of Arc with B as starting point

Step 5

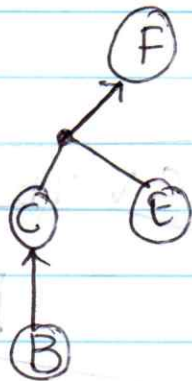So next we will see for the Rule that result in F being True
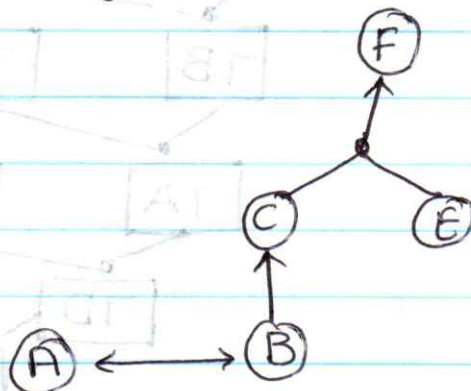

Now we add C & E to the stack
E is already True
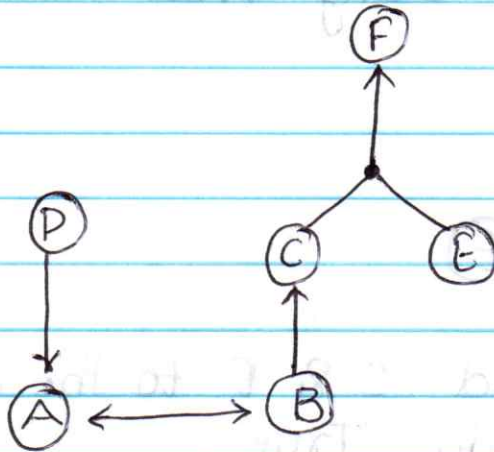We will Look for result that make C true
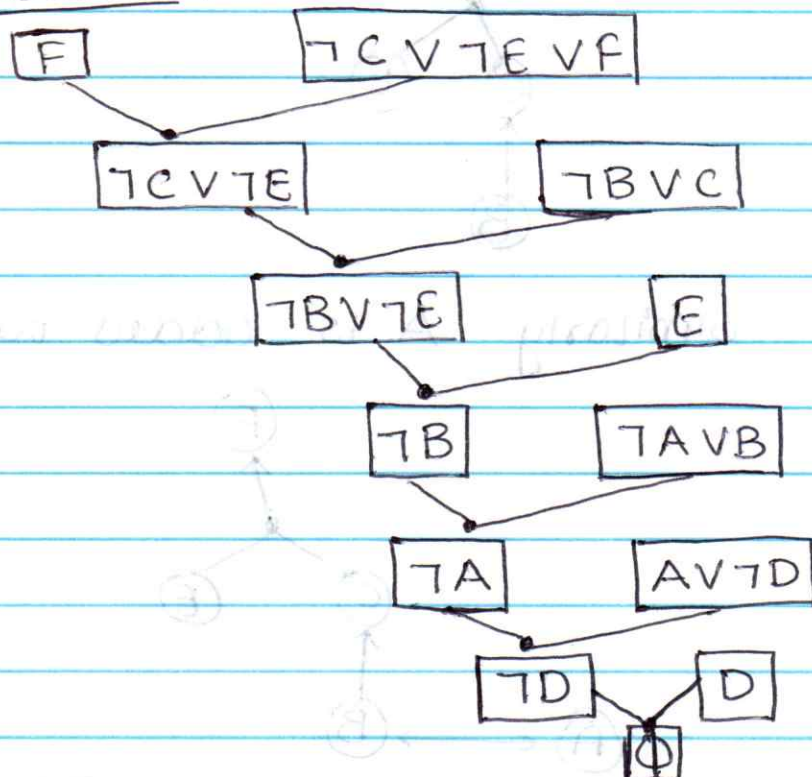
So now we add B to stack

Similarly A is added to stack

Now we add D to the stack.

Thus we can say "KB entails F"

iii) Resolution

4)   Task 4

ⓐ   Constants =   John, Mary, May 1 2017,
                     May 2 2017, $10,000
                     May 3 2017

Predicates =
• Rain(x) = Rained on day x
• give (x, y, z, m) = x must give a cheque
        of y to z on day m
• mow (x, y) = x mow the lawn of day y

• rain(May 1, 2017) = rain on May 1, 2017 = Ⓐ
• give (john, $1000, Mary, May 2 2017)
    =   John gives Mary a cheque of $10000
    on may 2 2017 = Ⓑ
• Mow (Mary, may 3 2017) = Mary mow lawn on
        May 3, 2017 = Ⓒ

Contract
               A → B
               B → C

ⓑ   What truly happend :-
                    ¬A ∩ B ∩ C

i.e $\neg$ rain (May 1, 2017)
given (John, $1000, Mary, May 2 2017)
Mow (Mary, May 3 2017)

c) No, the contract was not violated as
per FOL because A → B is always
true, & Event B & C took place
& event A did not occur

D) Taking constant & predicates from
part ⓐ

Contract

rain (May 1 2017) ⇒ give (John, $10000, Mary
May 2 2017)

give (John, $10000, Mary, May 2 2017)
⇒ mow (Mary, may 3 2017)

what truly happend

$\neg$ rain (May 1 2017) ∩ give (John, $10000, Mary
May 2 2017) ∩ Mow (Mary, may 3 2017)

Task 5

(5) <u>Predicates</u>

$Adult(x)$: x is adult
$Child(x)$: x is child
$Boat(x)$: x is boat
$Onleft(x)$: x is on left side.
$Onright(x)$: x is on right side.

<u>Initial state</u>

$Boat(b) \cap Onleft(b) \cap Child(c_1) \cap onleft(c_1)$
$Child(c_2) \cap onleft(c_2) \cap Child(c_3) \cap onleft(c_3)$
$\cap adult(a_1) \cap onleft(a_2) \wedge adult(a_2) \cap onleft(a_2)$
$\cap Adult(a_3) \cap onleft(a_3)$

<u>Goal state</u>

$Onright(b) \cap child(c_1) \cap onright(c_1) \cap child(c_2)$
$\cap onright(c_2) \cap child(c_3) \cap onright(c_3) \cap adult(a_1)$
$\cap onright(a_2) \cap adult(a_2) \cap onright(a_2)$
$\cap adult(a_3) \cap onright(a_3).$

Operation

Action : move right two (x, y, b)
precondition: Child(x) ∩ child(y) ∩ onleft (x)
    ∩ Onleft (y) ∩ onleft (b) ∩ boat(b)
Effects: Onright (x) ∩ onright (y) ∩ onright(b)
    ∩ not (Onleft (x)) ∩ not (onleft (y)) ∩ not(onleft(b))

Action: move right (x, b)
precondition: Onleft (x) ∩ onleft (b) ∩ boat(b)
Effects: Onright(x) ∩ onright(b) ∩ not(onleft(x))
    ∩ not (onleft (b))

Action : move right two (x y b)
precondition: child (x) ∩ Adult(y) ∩ onleft (x)
    ∩ Onleft (y) ∩ onleft (b) ∩ Boat (b)
Effects : Onright (x) ∩ onright(y) ∩ onright (b)
    ∩ not (onleft (x)) ∩ not(onleft (y)) ∩ (not (onleft(b))

Action: move left two (x y b)
precondition: child (x) ∩ child(y) ∩ onright(x) ∩ onright (y)
    ∩ (onright (b) ∩ Boat (b)
Effects : onleft (x) ∩ onleft(y) ∩ onleft (b) ∩ not(onright(x))
    ∩ (not(onright (y)) ∩ not (onright (b))

Action: move lyt (x b)
precondition: Onright (x) ∩ Onright(b) ∩ Boat(b)
Effect: Onleft (x) ∩ Onleft(b) ∩ not(onright(x)) ∩ not(onright(b))

Action: move left two (xyb)
precondition: child(x) ∩ adult(y) ∩ onright(x) ∩ onright(y)
   ∩ Onright(b) ∩ boat(b)
Effect: Onleft(x) ∩ onleft(y) ∩ onleft(b) ∩ not(onright(x))
   ∩ (not(onright(y))) ∩ not(onright(b))

Complete Plan

    move left two (c_1, c_2, b)
    move right (c_1, b)
    move lyt (c_1, b)
    move right two (c_1, a_1, b)
    move lyt (c_1, b)
    move right two (c_1, a_2, b)
    move lyt (c_1, b)
    move right two (c_1, a_3, b)
    move lyt (c, b)
    move right $(c_1 c_3 b)$

⑥   Task 6

In JUNGLE World there are
     4 predicates
     4 arguments
     5 constants.

4 predicates take [1 4] arguments.
Number of ways to assign 5 constant
$$\begin{bmatrix} 4 \times 5^1 & 4 \times 5^4 \\ 20 & 2500 \end{bmatrix}$$

The PDDL state is defined by Listing all the
predicates that are true

For n predicates, the possible states are

$$nc_0 + nc_1 + nc_2 \cdots nc_n = \sum_{i=0}^{n} nc_i = 2^n$$

∴ tight bound on the number of
state in JUNGLE World is

$$\begin{bmatrix} \sum_{i=0}^{20} 20c_i & \sum_{i=0}^{2500} 2500c_i \end{bmatrix}$$

$$= \begin{bmatrix} 2^{20} & 2^{2500} \end{bmatrix}$$

7) Task 7

- Execution Monitoring | Online Replanning :-
  For this there is no need to make any modification as it would replan the entire scenario.

  If the goal of an action is not satisfied then the system replans the flow again from the current state

- Conditional planning :-

  For this we need to modify

Action : move right (x, b)
Pre Condition : Onleft $(x) \cap$ Onleft $(b) \cap$ Boat (b)
Effect : (Onright $(x) \cap$ Onright $(b) \cap$ not (Onleft (x)) $\cap$ (not (Onleft (b)))) $\vee$ (Onleft $(x) \cap$ Onleft (b))

Action : move left $(x b)$
pre condition : Onright $(x) \cap$ onright $(b) \cap$ boat (b)
Effect : (Onleft $(x) \cap$ Onleft $(b) \cap$ not (Onright (x)) $\cap$ not (Onright (b))) $\vee$ (Onright $(x) \cap$ Onright (b)))