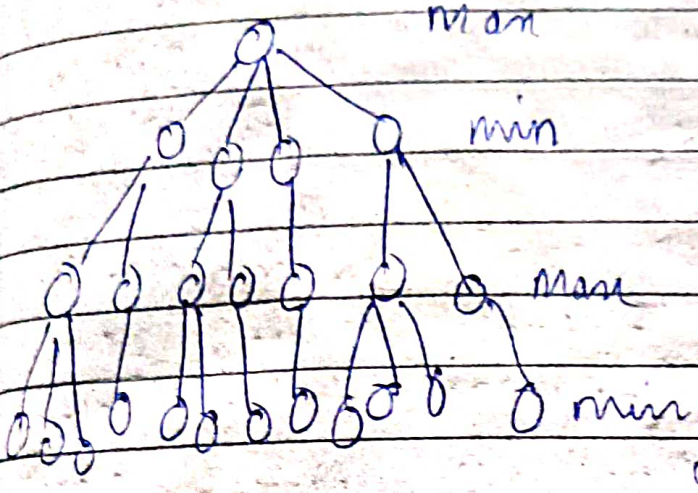
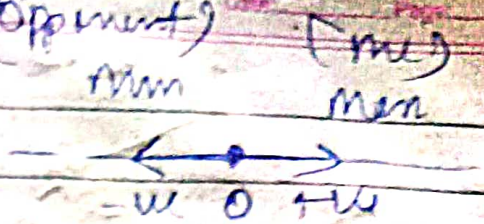
12.09.2023
Tuesday





deep-enough = T

evaluation function

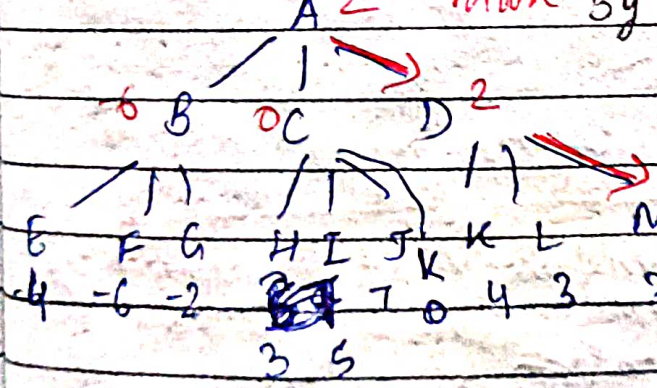
cutoff branch initially
- α-cutoff
- β-cutoff

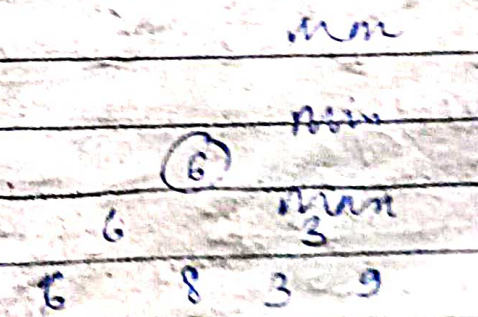
↳ search would be reduced.

lyt-right α-β cutoff
right-lyt α-β cutoff.

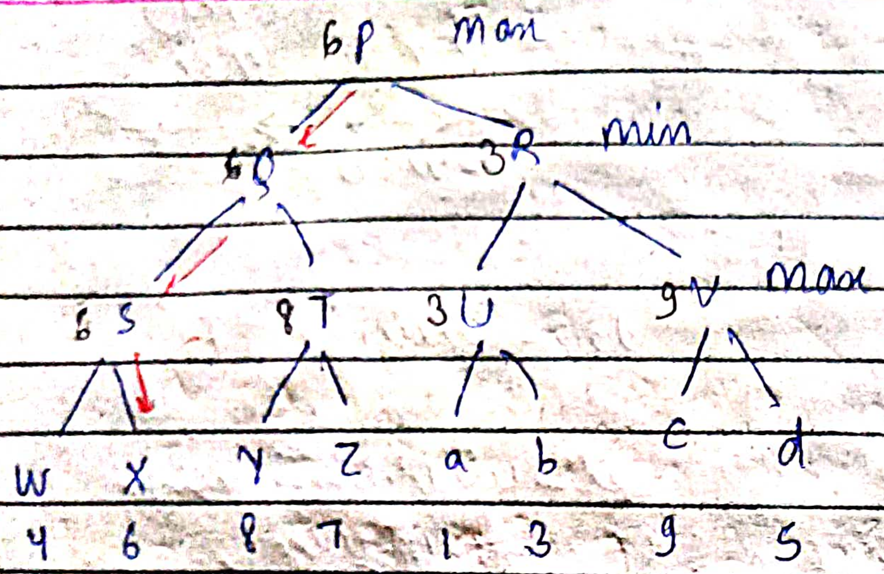
Consider the following tree. Find out value of root using mini-max search algo.

By default (max)





P = 6



β will come at minimizing level.

β — min — upperbound

β cutoff / β pruning

α — max — lower-bound

α cutoff / α pruning

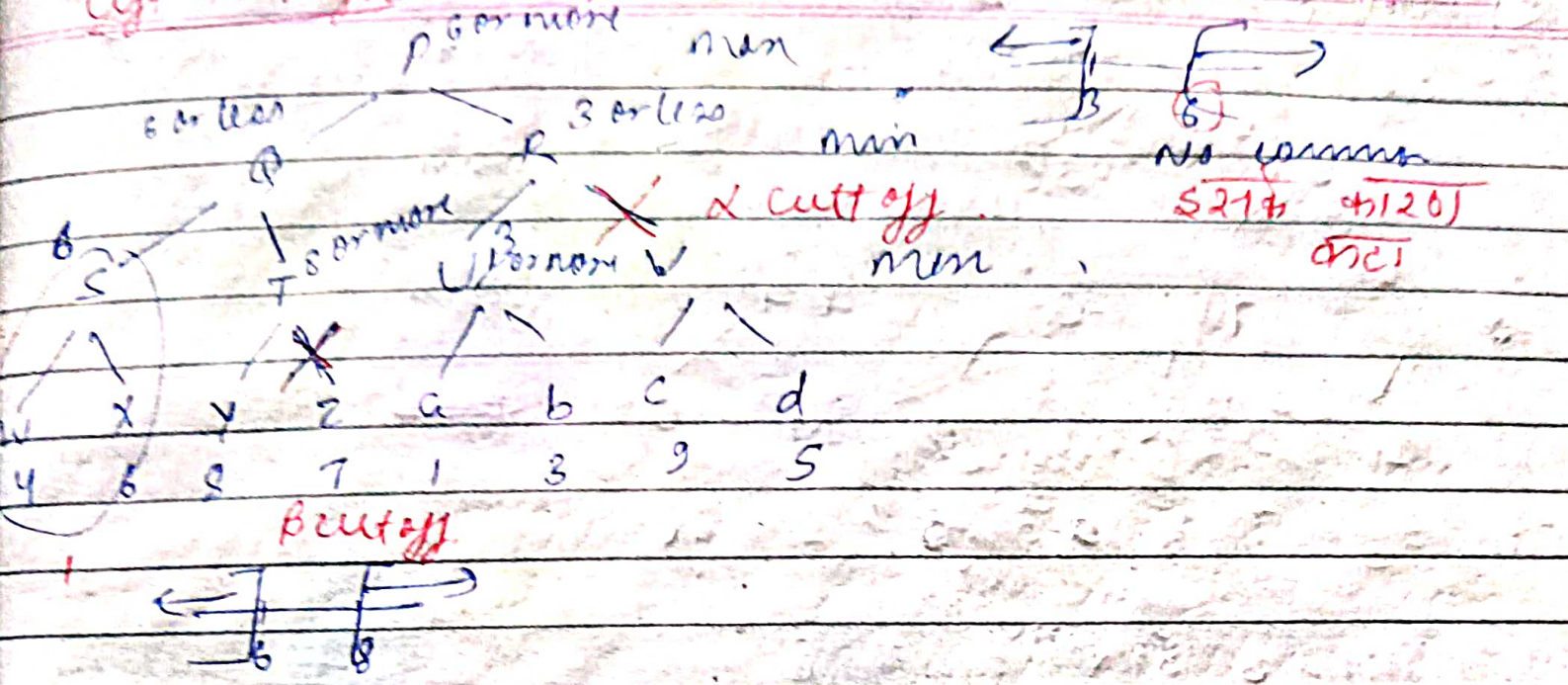
① lower bound at minimizing level is α value

② cutoff occurring due to α value is α - cutoff.

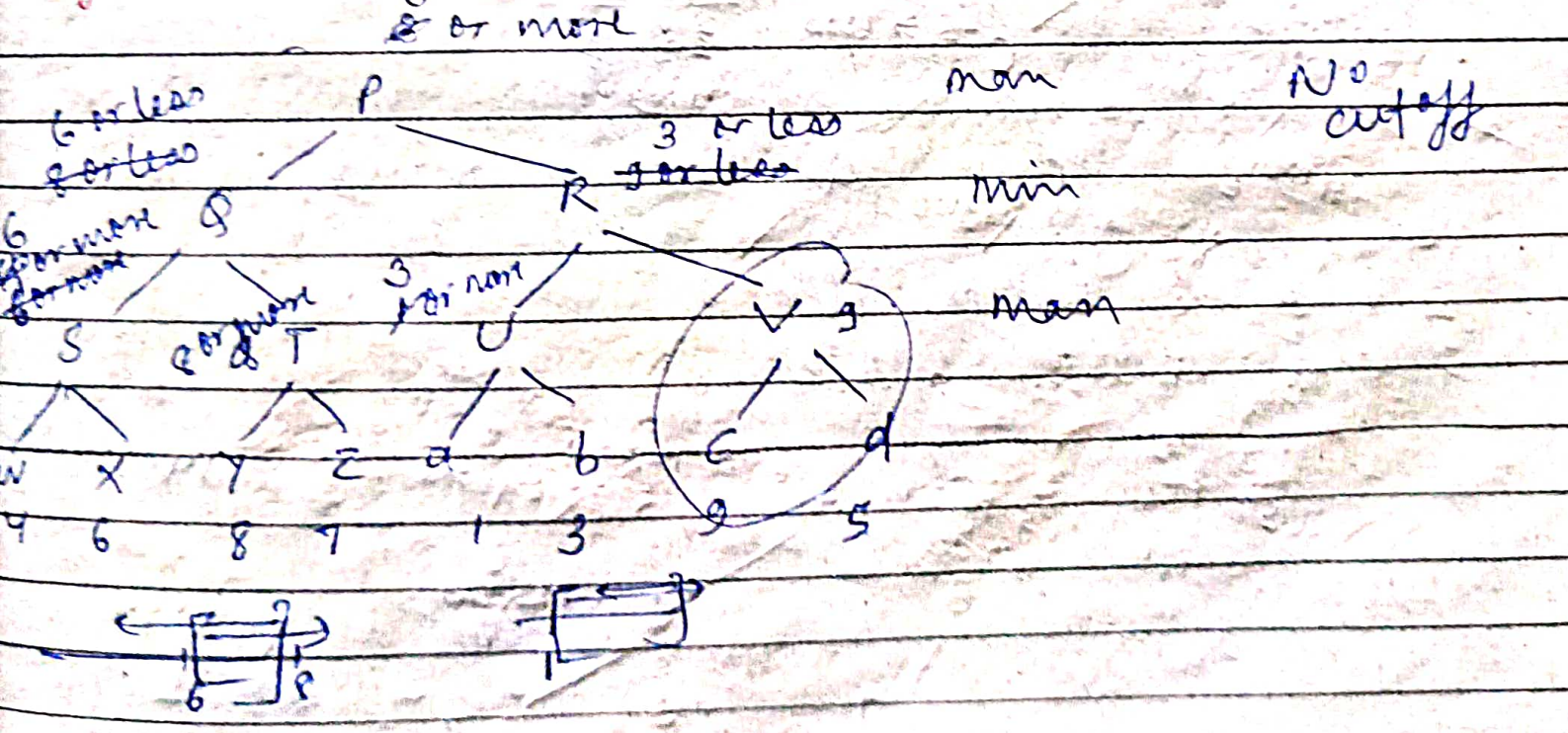
Same for β.

Range line वाले के चक्र |

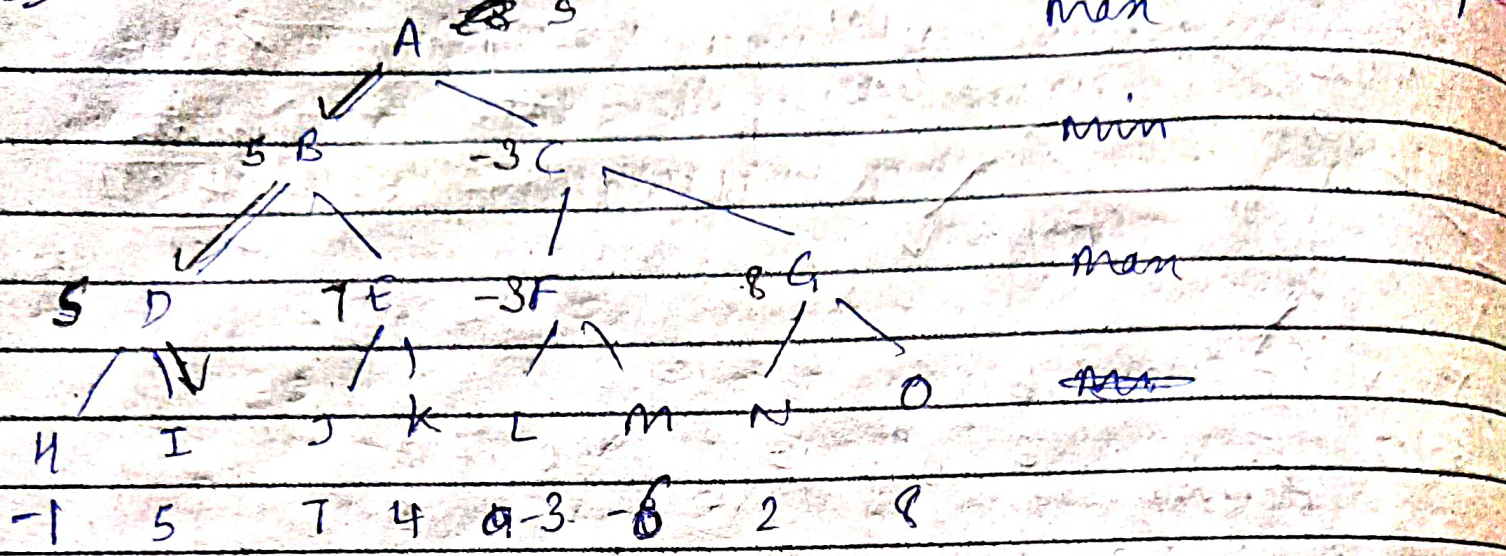
Left to Right α - β



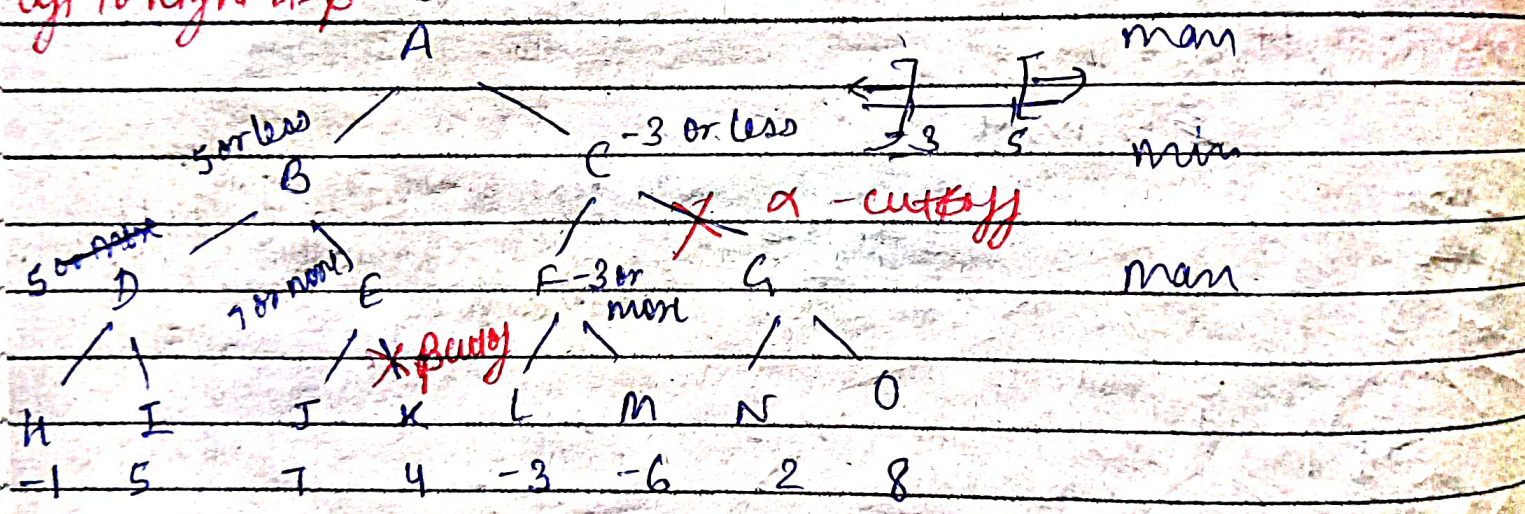
Right to Left:



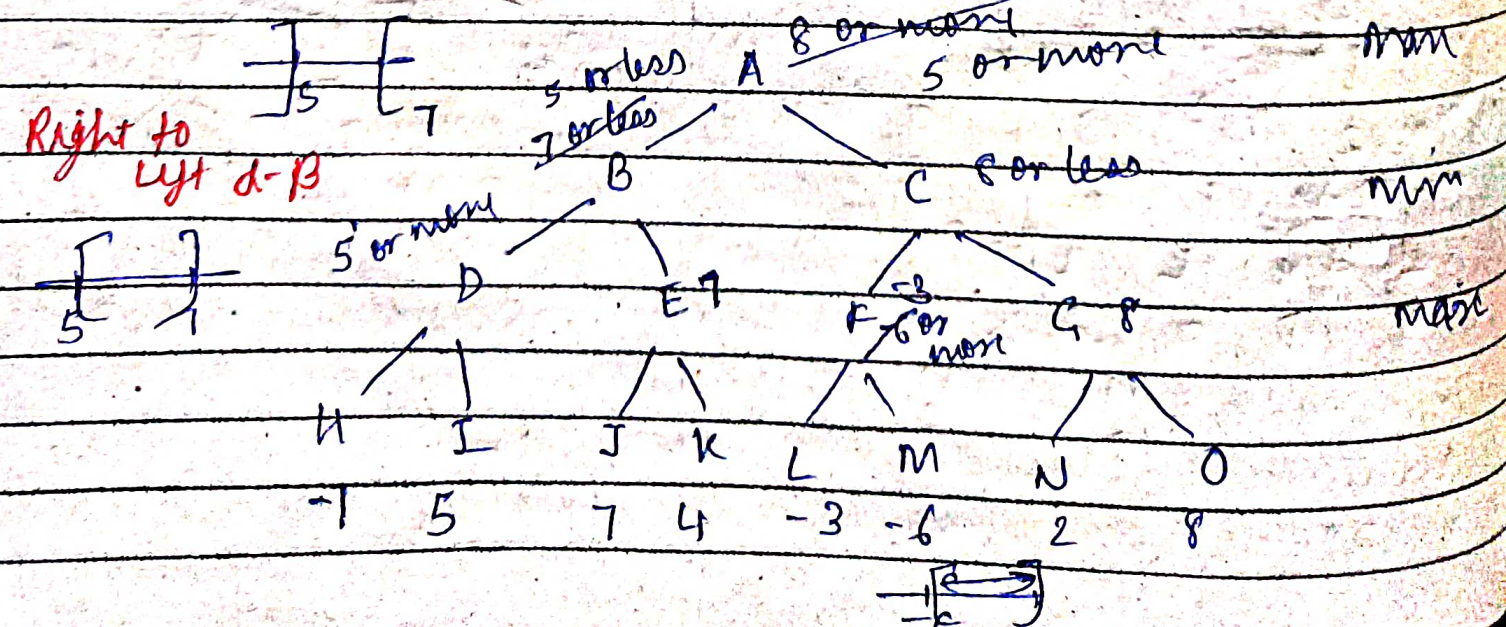
Q) Find Root value, & Lft to Right α-β, Right to Lft α-β



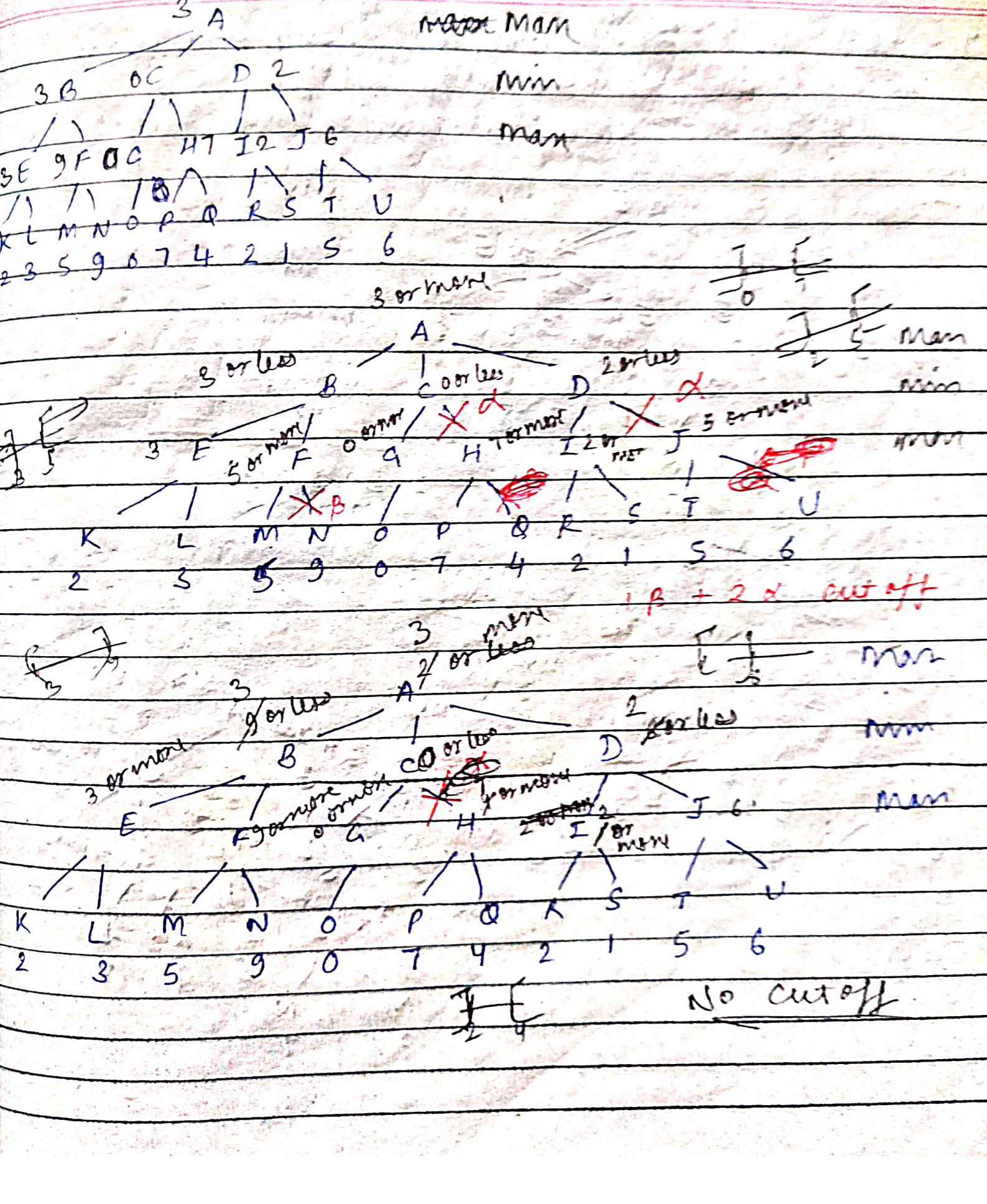
Lft to Right α-β



Right to Lft α-β



8 puzzle using α-β
Tic Tac Toe using α-β



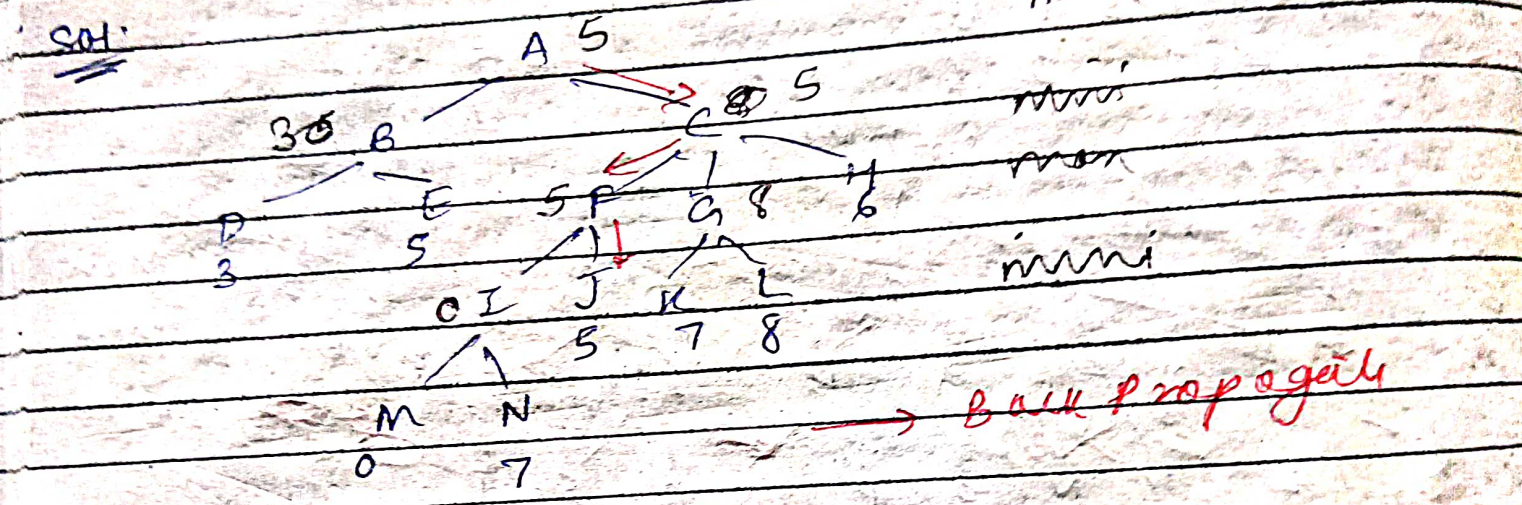
14.02.2025
Thursday

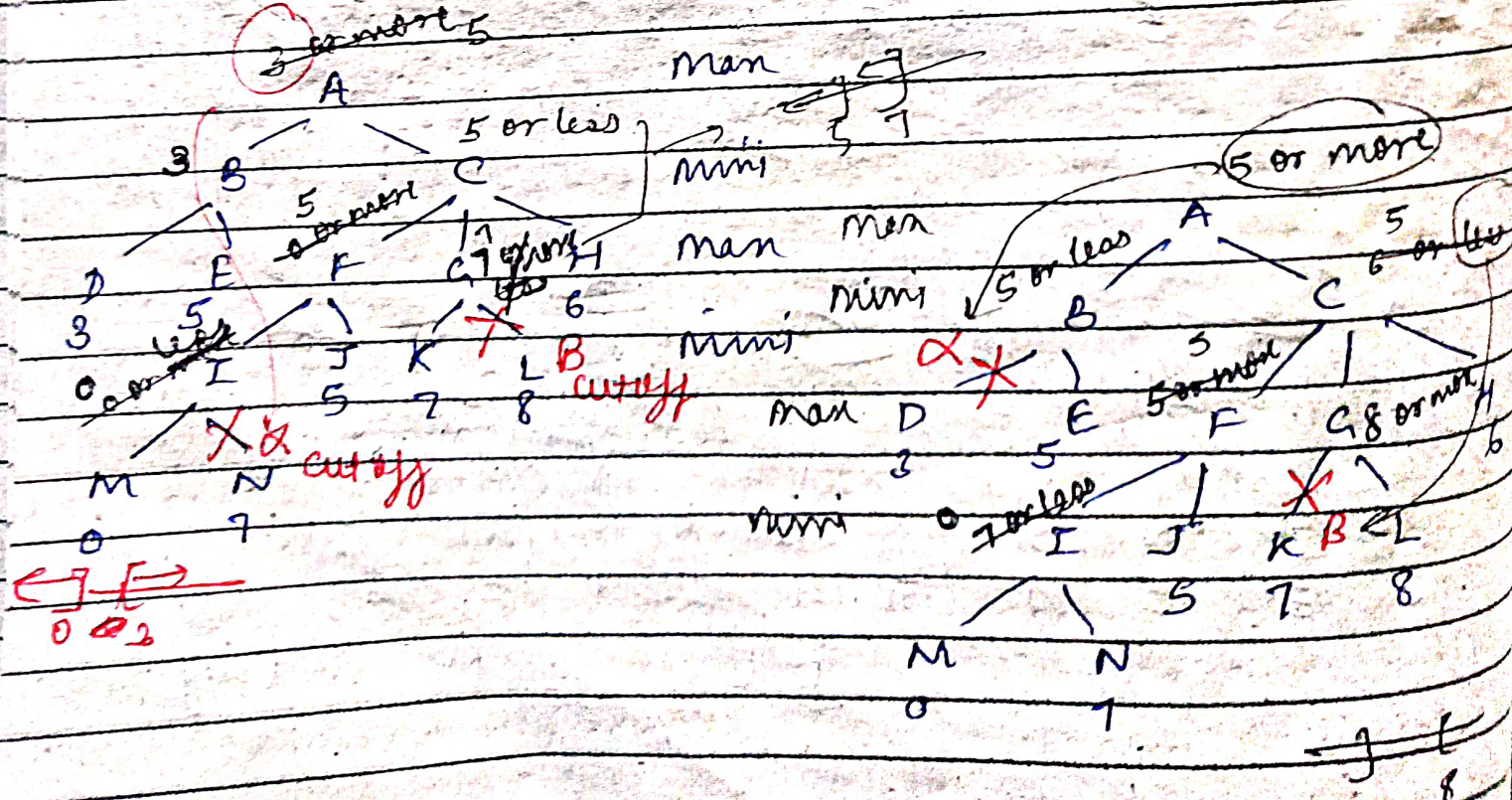
max. Node 42 ∝

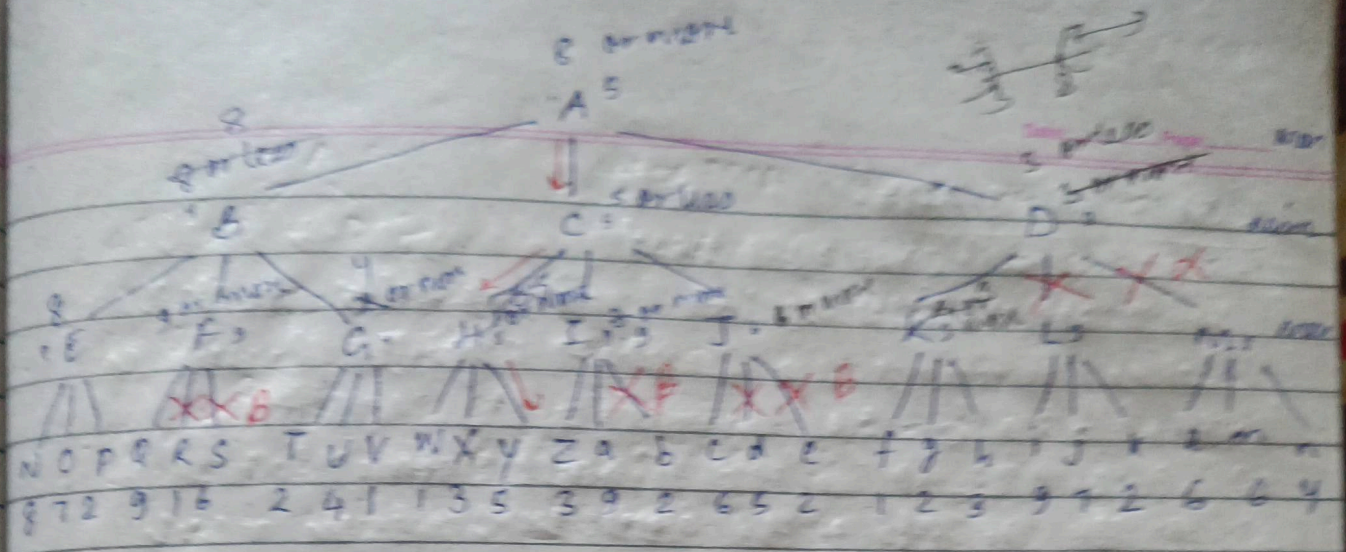
Date: Page:

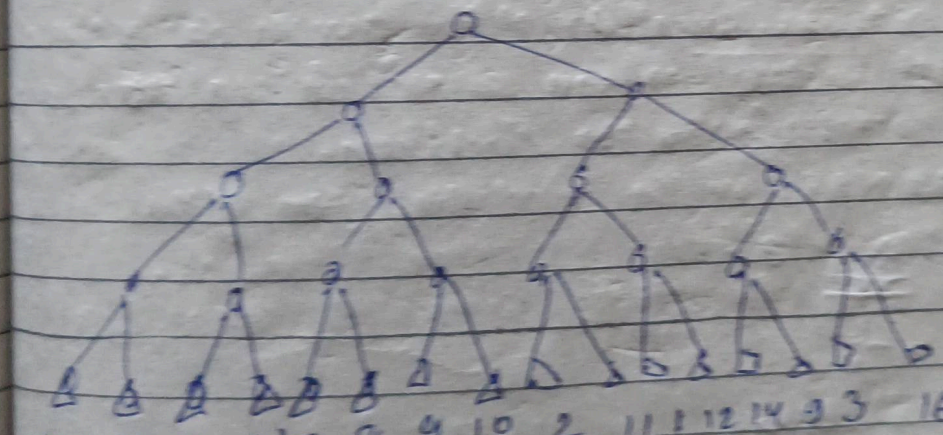
Q) Consider the following game tree Using min max find. ① Left to Right ∝-β cutoff.
② Using Right to Left ∝-β pruning (cutoff)

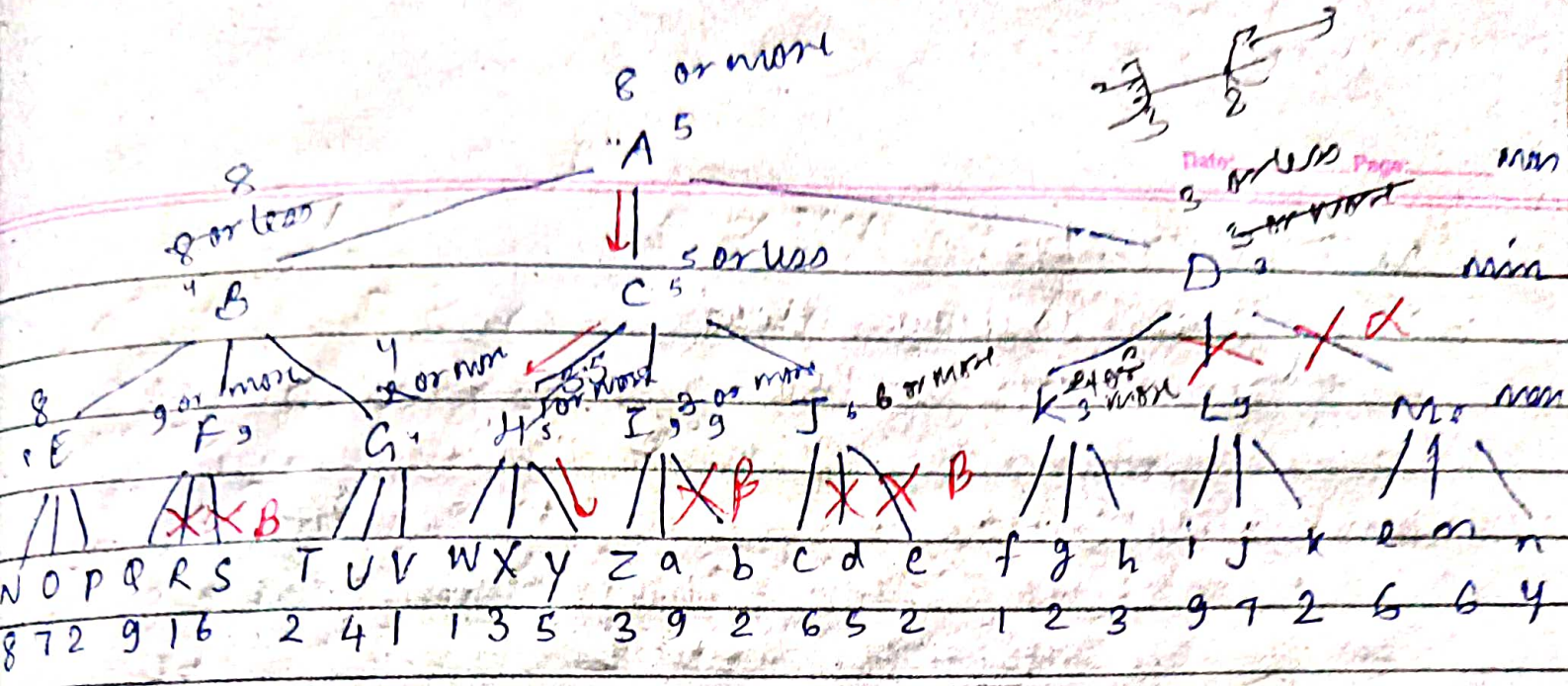
Sol:







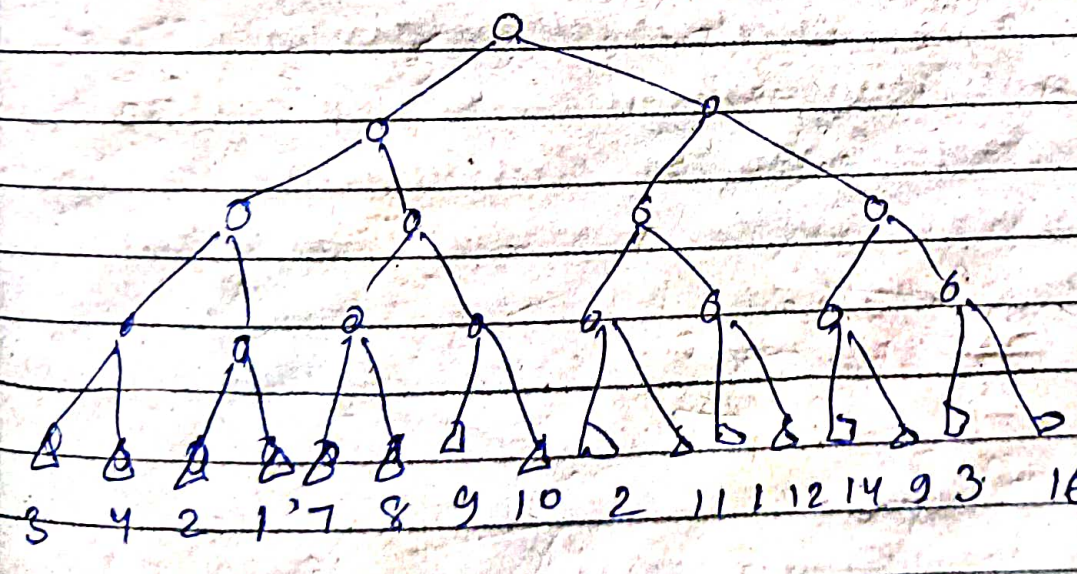




Backtrack → A → C → H → Y

Ib, Jd, Je } B

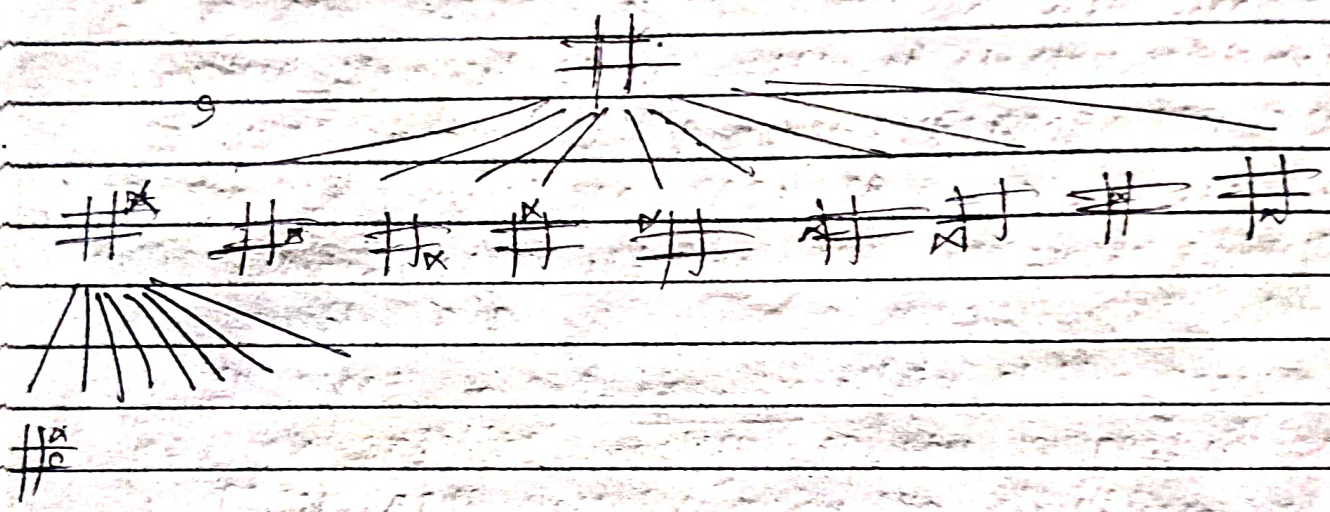
Dl, Dm } d

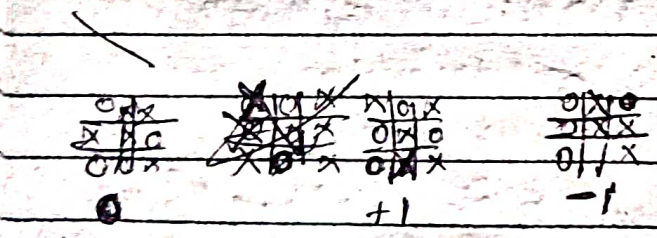


Date: Page:

Back Tracking Algo.

- Best move strategy used
  - Max will try to maximize its utility. (Best move)
  - Min will try to minimize its utility (Worst move).





No one wins. I wins (1st state) II wins.

Plausible Move Generator → All possible children are generated.

Static Evaluation function → Evaluate value of the path.
(Utility value)

20.03.2023

Rich & Knights

Date: Page:

MinMax :-

(1) Set FINAL_VALUE to be as minimum as possible.
(2) If limit of search has been reached then FINAL_VALUE = GOOD_VALUE of the current position.
(3) Else do
  1. Generate the successor of the position
  2. Recursively call MINMAX again with the present position with depth incremented by UNITY.
  3. Evaluate the GOOD_VALUE. → minimizer ⇒ at minimum
(4) If GOOD_VALUE > FINAL_VALUE then update FINAL_VALUE ⇒ maximizer ⇒ at maximum

MIN MAX :-

(1) If the determine if the limit of search has been reached or if the level is minimizing level or if the level is maximizing level.
(a) If the limit of the search has been reached compute the static values of the current position relative to the appropriate player. Report the result. (Leaf Nodes ⇒ value for all)
(b) If the level is minimizing level, use MINIE-MAX on the children of current position. Report the minimum of result. (minimum value)
(c) If the level is maximizing level, use Mini-Max on the children position. Report the maximizing of result. (maximum value)

α-β Cut off :-

(1) Determine if the level is top level or if the limit of the search has been reached or if the level is minimizing level or if the level is maximizing level.

(a) If the level is top level, let $\alpha = -\infty$ & $\beta = +\infty$

(b) If the limit of the search has been reached compute the static values of current position relative to the appropriate player. Report the result.

(c) If the level is MINIMIZING level.

(i) Until all children are examined with MINIMAX or $\alpha \geq \beta$

(A) Set β to the smaller of the given β value & smallest value so far reported by MINIMAX working on the children.

(B) Use MINIMAX on the next children of the current α & β.

(ii) Report β.

(d) If the level is MAXIMIZING level.

(i) Until all children are examined with MINIMAX or $\alpha \geq \beta$

(A) Set α to the larger of the given α-value & biggest value so far reported by MINIMAX working on the children.

How can we extend MIN-MAX. to more than 2 players.

AI engineers is knowledge engineers - we read simulation.

(B) Use MINIMAX on the next child of the current position & handling this new application of MINIMAX on the current α & β.

(ii) Report α.

Q. What is the reason, why game playing plays a major role in AI?

① The rules of the games are limited. Enhancing knowledge of game specific.

② Many humans assist in developing of problem

Knowledge Engineers → goto (knowledge acquires) → multiple Domain experts (Chess players, internet)

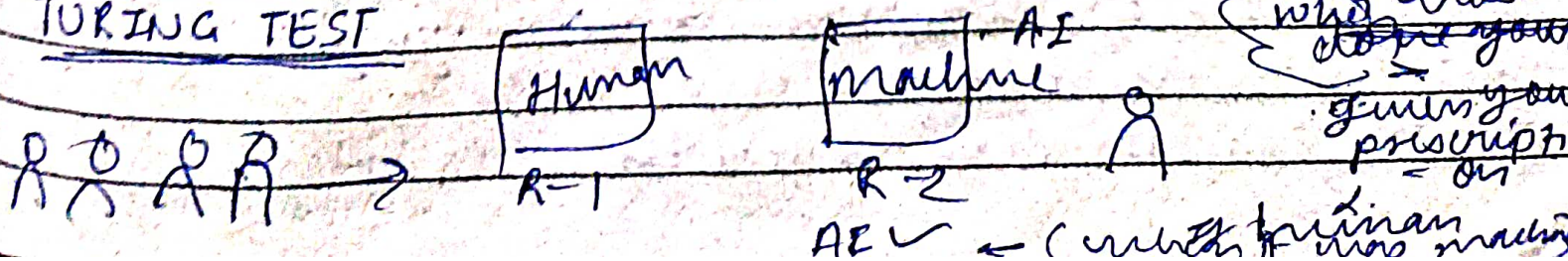
Production rules are created.

LHS → RHS

(many to many)

Expert system (only when machine can replace human)

TURING TEST



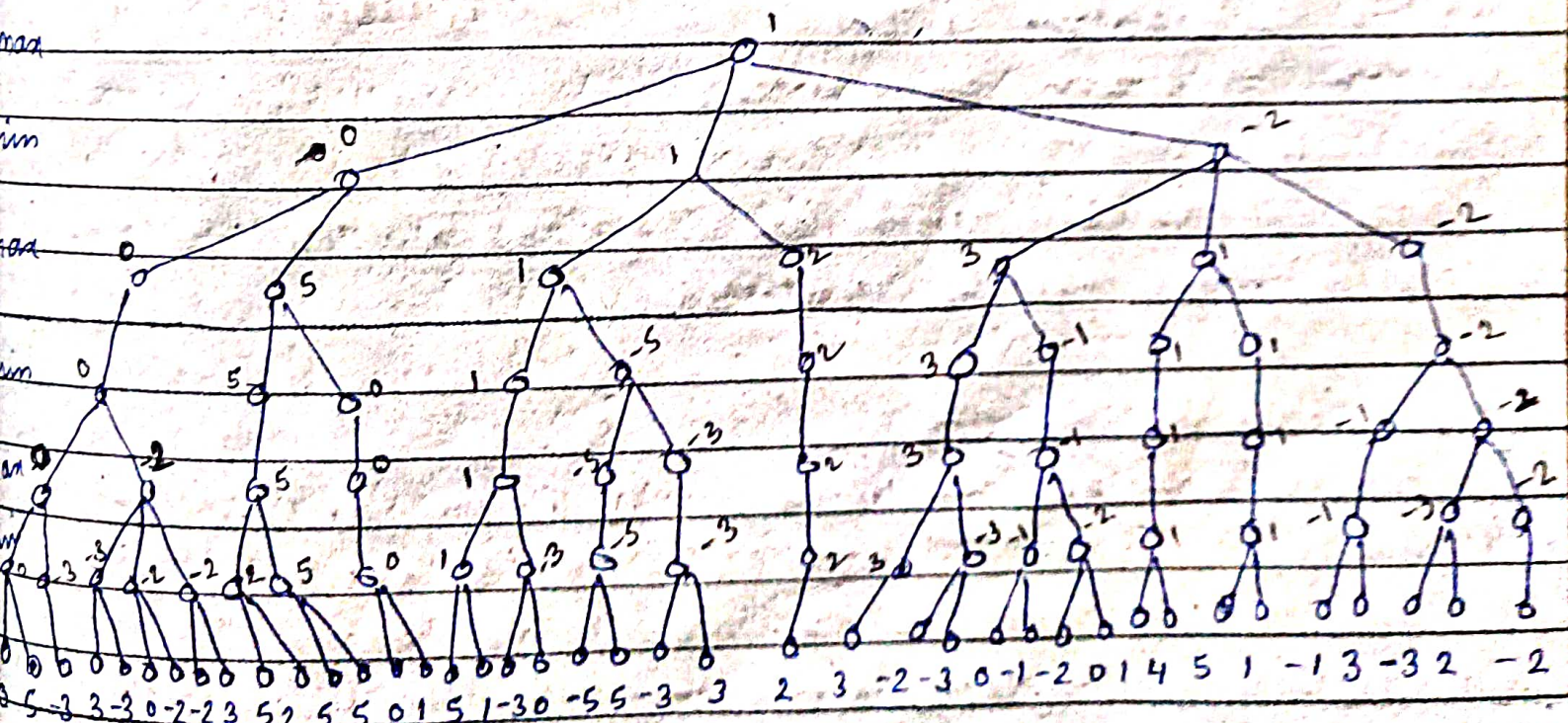
Date: Page:

③ Games provide a
① The rules of the game are limited. Hence
② extensive amt of domain specific knowledge are needed.
④ many human experts exist to assist in the development of the program. Hence the problem of having human experts does not arrises.
③ Games provide a structured task where result success or failure of can be measured with less effort.
④ For the human expert, it is easy to explain the rationale of for a move unlike other domains → reason
⑤ Games pictures real life situation in a constricted manner fashions. The logical reasoning ability of the human in a normal condⁿ & under stress is clearly exhibited in game playing. Game playing permits one to simulate real life situations.
(IOT devices use of AI)
(Medical + use of AI)
⑥ ~~Con~~.

Q) Problems of computer game playing.
Sol: ① Horizon Effect.
② Optimal move question.

(1) Horizon Effect :- The computer chess playing programs have to be dealt with the major problem known as Horizon effect. These 2 programs do fixed depth search, if it is possible for program to postpone the danger by dragging the situation into the branch of the tree which the algorithm explores than the situation is out of the view of computer program or horizon.

(2) Optimal Move Question :- Second major effect is that these program expect the opponent to make most optimal move which can't be expected in real.



21.09.2023
Thursday

Date: Page:

```
  P A S
+ R B Q
-------
  S B K W
  A S A A  OK
  S E P B  OK  OK
-----------------
  S B S K A W
```

K + A = A
K = 0
B + A + B = K → 0 or
2B + A = 0 or 10 or 20

Date: Page:

PRODUCTION SYSTEM: (PS)

(Search system according to AI)

PS in AI helps in structuring AI programs that facilitates describing & performing the search process.

PS consists of
1. Set of Rules ← LHS → RHS (Forward Strategy search)
2. Knowledge Base & Database in AI
3. Control Strategy & When more than 1 rule match then which to take?
   - Perform the first
   - Perform the best
   - Perform the most recent (i.e. Recent data updated.)
4. Rule Applier.

Steps required to solve the problem –

1. Reduce the problem so that it can be shown in a precise statement
2. Problem can be solved by searching a path thro space
3. Solving process can be modelled.

Advantages of PS :–

1. Structuring AI problem (PS is an excellent tool)
2. Highly modular (new rule can be added easily)
3. Rules are expressed in its own natural form.

Characteristics of control Strategy :–

1. It should cause motion
2. It should be systematic.

Date Page

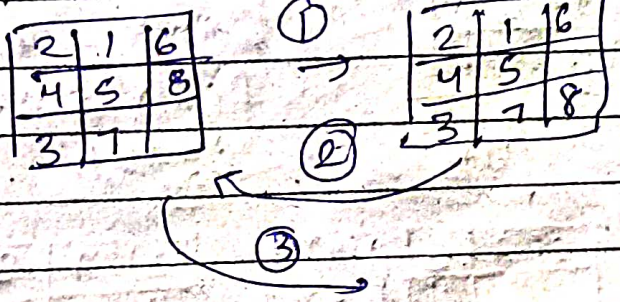
Classes of PS / Types of PS / Characteristics of PS :-

1. Monotonic P.S.
2. Non-Monotonic P.S.
3. Commutative P.S.
4. Partially Commutative P.S.

1. Monotonic PS (System which never prevents backtrack)

MPS is a PS in which application of a rule never prevents the later application of another rule which could be applied at the time the first rule was selected

8 Puzzle



2. Non Monotonic PS (prevents backtrack)

NPS is a PS in which upper one true.

Chess game more Backtrack etc.

4. Partially Commutative PS : (PCPS)

PCPS is a PS with a property that if the application of a particular sequence of rule, transform state X into Y. then any transform of permutation of those rules that are allowable also transform state x into state Y

x → y

1 4 3 2
4 1 2 3 } all permutation give same results.

03.10.2023
Tuesday

Date: Page:

③ Commutative PS : (CPS)
Monotonic + PCPS

8 PUZZLE PROBLEM (A* Algo)

$f(n) = g(n) + h(n)$

I.S.

2	8	3
1	6	4
7		5

G.S

1	2	3
8		4
7	6	5

2	8	3
1	6	4
7		5

$g(n) = 0$
$h(n) = 4$
$f(n) = 0 + 4 = 4$

up | left | right

Take minimum

2	8	3
1		4
7	6	5

$g(n) = 1$
$h(n) = 3$
$f(n) = 4$

2	8	3
1	6	4
7		5

$g(n) = 1$
$h(n) = 5$
$f(n) = 6$

2	8	3
1	6	4
7	5	

$g(n) = 1$
$h(n) = 5$
$f(n) = 6$

up | left | right

2		3
1	8	4
7	6	5

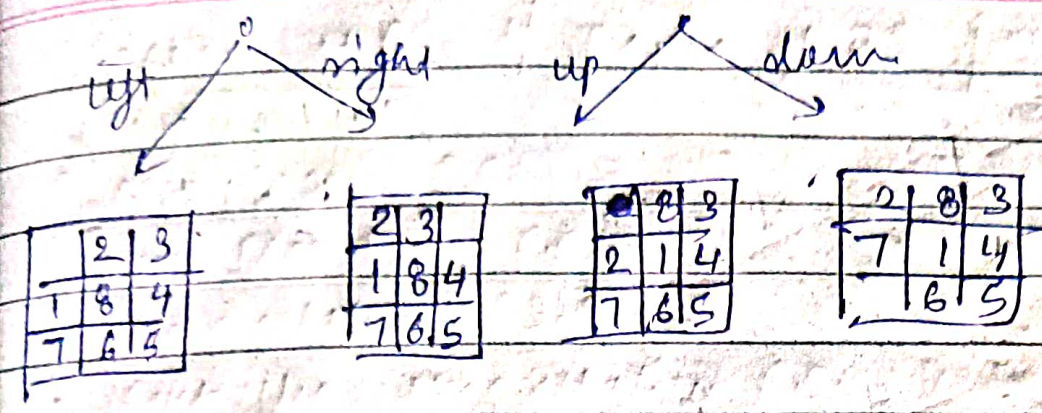
2	8	3
	1	4
7	6	5

2	8	3
1	4	
7	6	5

$g(n)$	= 2	2	2
$h(n)$	= 3	3	4
$f(n)$	= 5	5	6

Can't both
(Least) Take both.

Date: Page:



	2	3
1	8	4
7	6	5

2	3	
1	8	4
7	6	5

	8	3
2	1	4
7	6	5

2	8	3
7	1	4
	6	5

$g(n) = 3$ 3 3 3

$h(n) = 2$ 4 3 4

$f(n) = 5$ 7 6 7

down

1	2	3
	8	4
7	6	5

$g(n) = 4$

$h(n) = 1$

$f(n) = 5$

Can explore other path if $f(n)$ less for them.

down right

1	2	3
7	8	4
	6	5

1	2	3
8		4
7	6	5

G.S.

$g(n) = 5$ 5

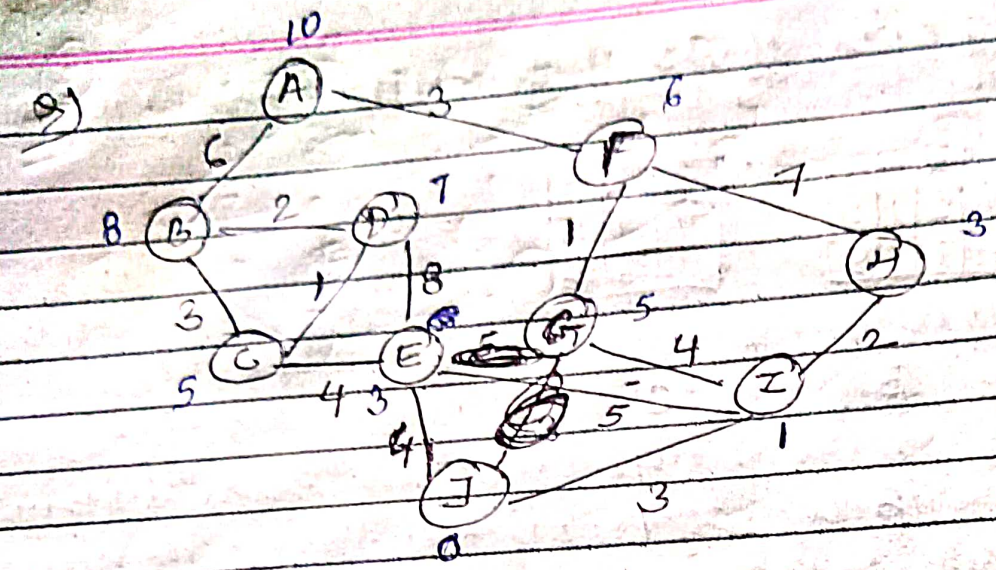
$h(n) = 2$ 0

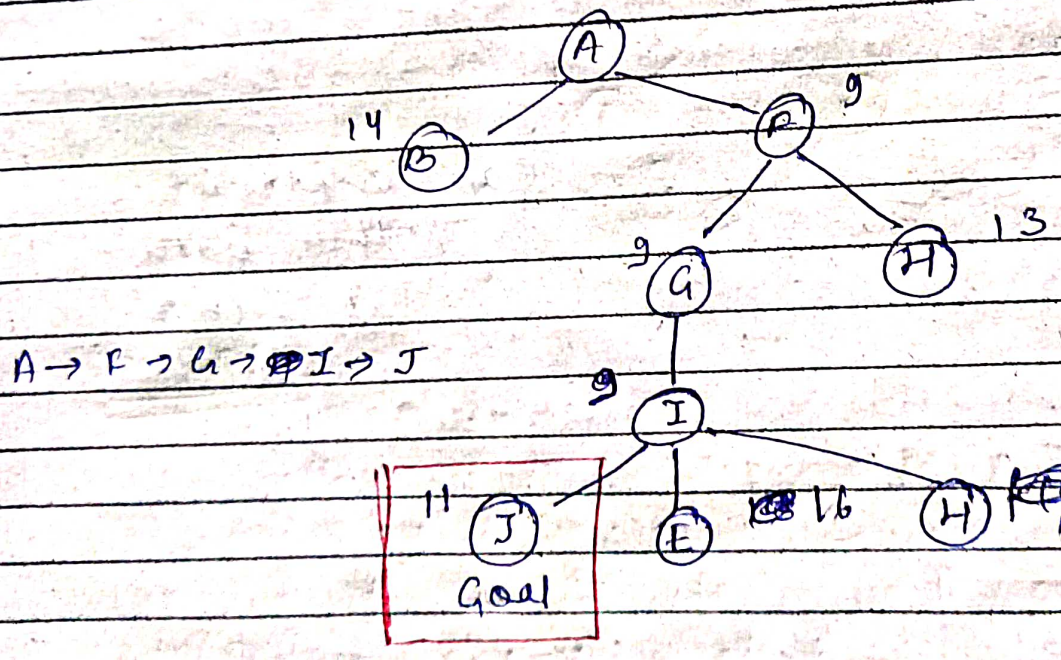
$f(n) = 7$ 5

If start & Goal Node is not given { S : Start Node, G : Goal Node } If G) = 0

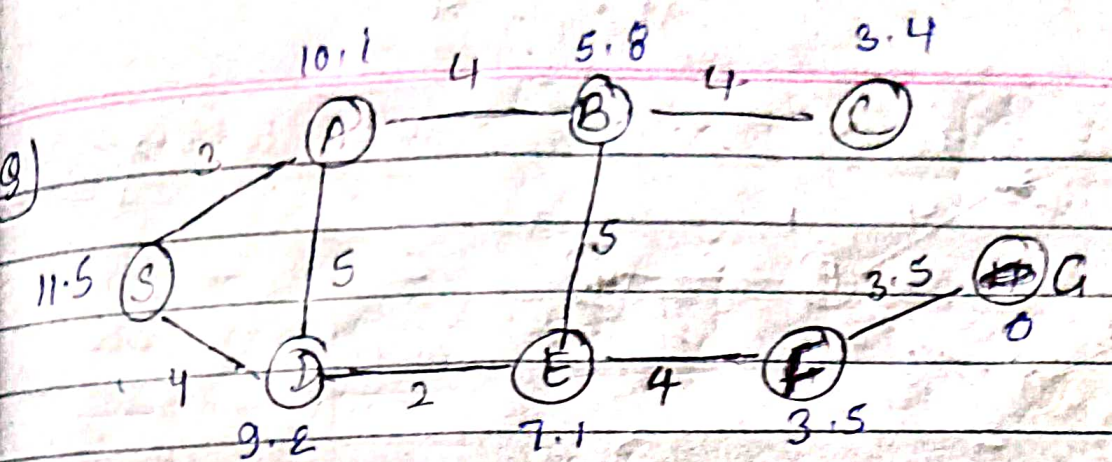
Date: Page:

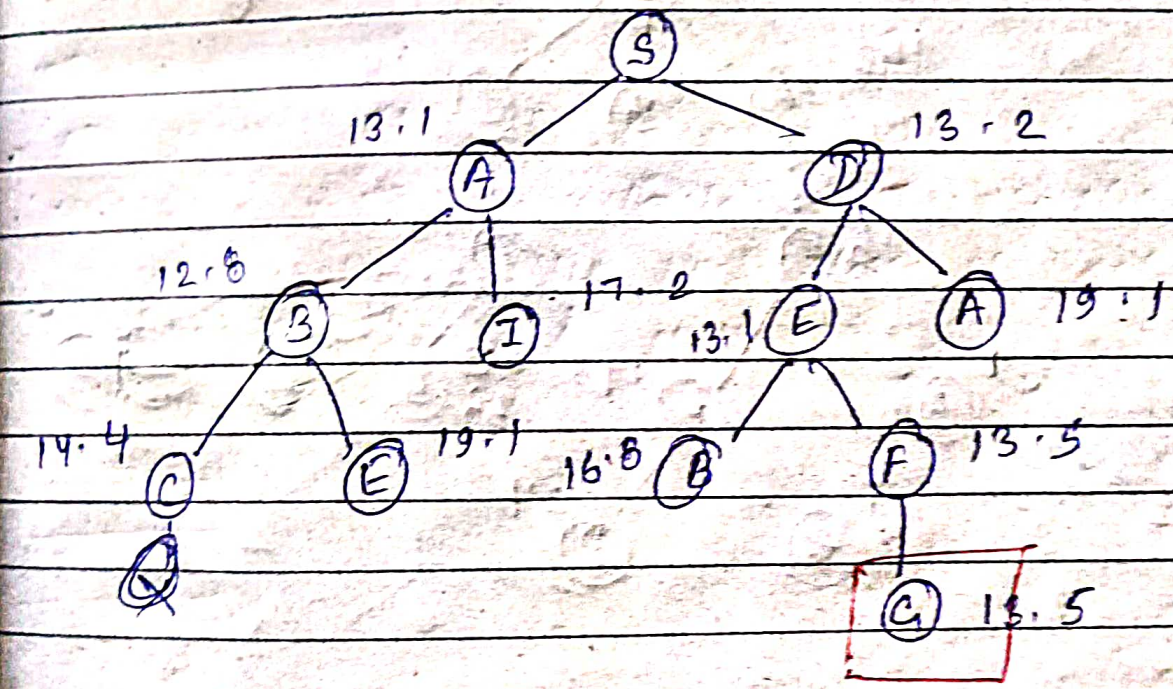
Start node : A
Goal node : J



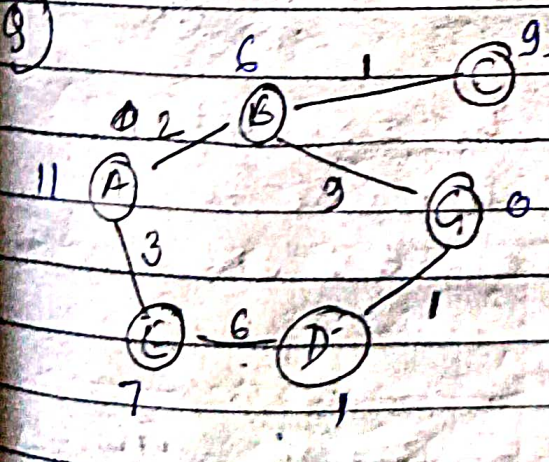


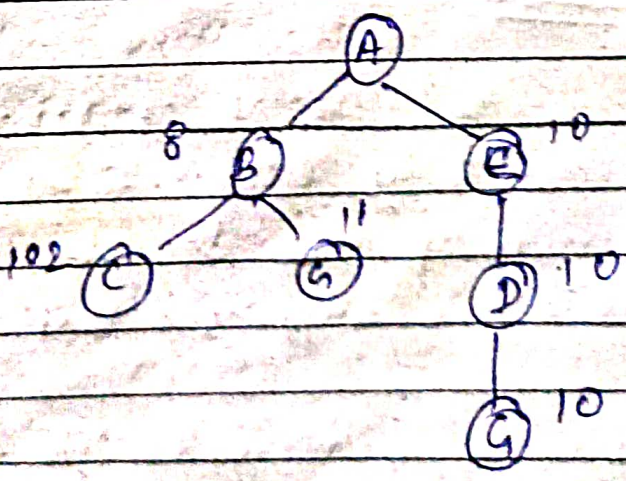
A → F → G → I → J


10.1
4
5.8
4
3.4
A
B
C
2
11.5
S
5
5
3.5
G
0
4
D
2
E
4
F
9.2
7.1
3.5


S
13.1
A
13.2
D
12.6
B
I
17.2
13.1
E
A
19.1
14.4
C
E
19.1
16.8
B
F
13.5
G
13.5

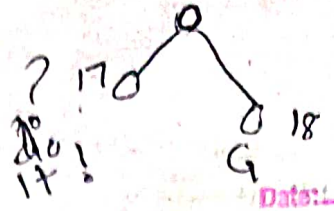
S → D → E → F → G


6
1
99
B
2
11
A
9
0
3
6
1
C
D
7
1

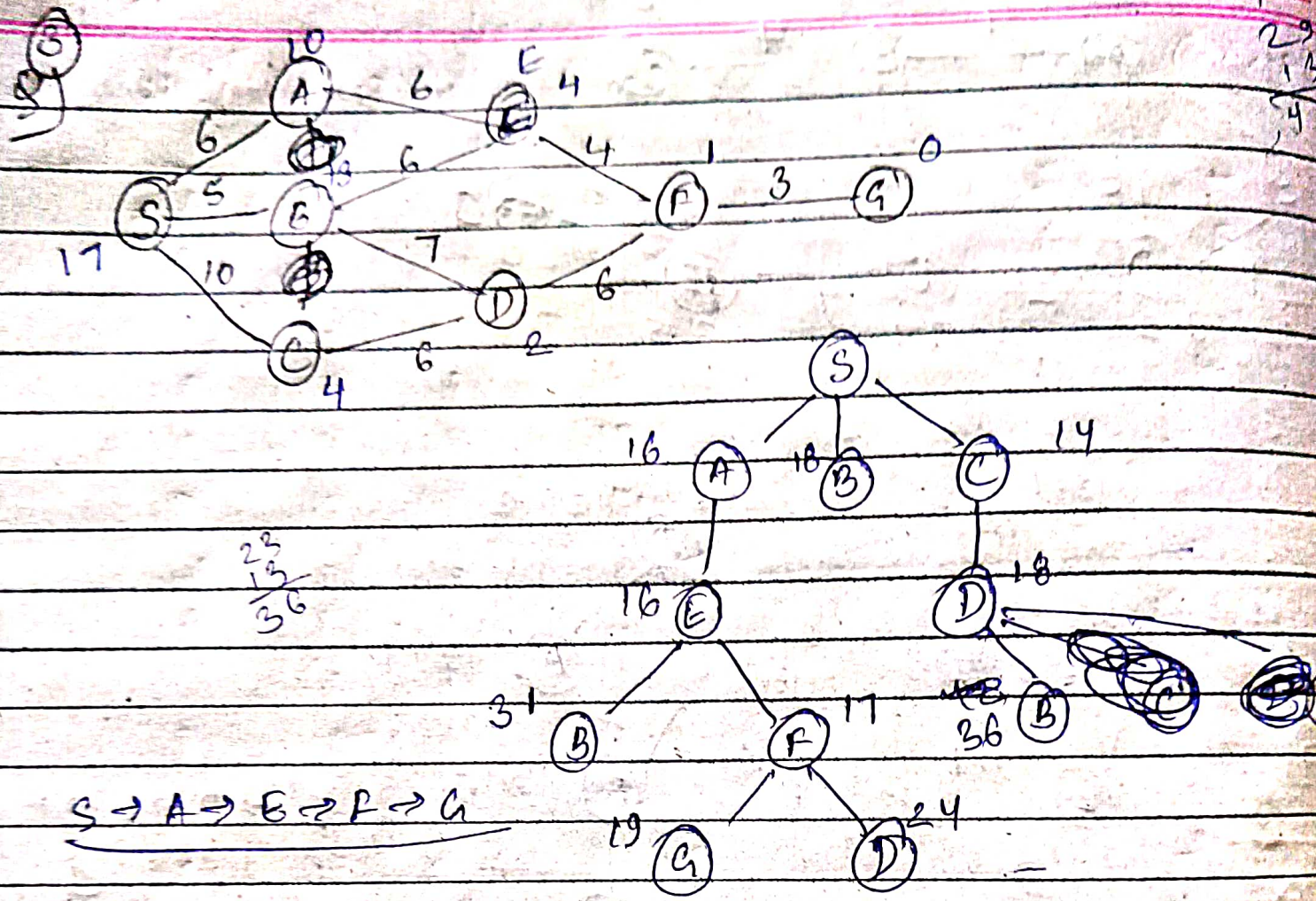

A
8
10
B
E
102
11
D
10
10

A → E → D → G

03.10.2023
Thursday



Date: Page:



S → A → E → F → G

A* Algo

(1) Start with OPEN containing only the I.S. Set that node's g value to 0 & its h value to whatever it is & its f = h + 0 = h. Set CLOSED to empty list.

(2) Until a goal node is found, repeat the follo
If there are no nodes on OPEN report fail
FAILURE
Otherwise pick the node on OPEN with th

least
~~best~~ f value. Call it as BESTNODE, Remove it from OPEN & place it on CLOSED. See if the BESTNODE is a goal node. If so exit & report sol^n Otherwise, generate the successors of BESTNODE.

For each successors do the following :—

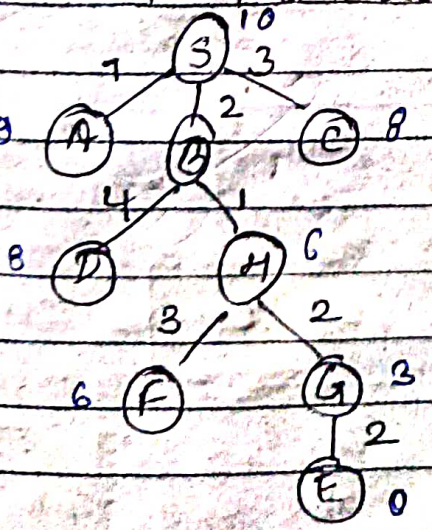
a) ~~SET~~ SET SUCCESSOR to point back to BESTNODE. This backwards link will make it possible to recover the path once a sol^n is found.

b) Compute g(SUCCESSOR) = g(BESTNODE) + the cost of getting from BESTNODE to SUCCESSOR

c) Compute f(SUCCESSOR) = g(SUCCESSOR) + h(SUCCESSOR)

d) if SUCCESSOR was not already at either OPEN or CLOSED, then put it on OPEN & add it to the list of BESTNODE's SUCCESSOR

Q)



OPEN

Node	g(n)	h(n)	f(n)
S	0	10	10
A	7	9	16
B	2	7	9 ✓
C	3	8	11
A	7	9	16
C	3	8	11
D	6	8	14
H	(3)	6	9 ✓
A	7	9	16
C	3	8	11
D	6	8	14
F	6	6	12
G	(5)	3	8 ✓
A	7	9	16
C	3	8	11
D	6	8	14
F	6	6	12
E	7	0	7 ✓

Goal Node

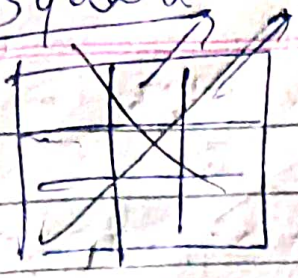
CLOSED

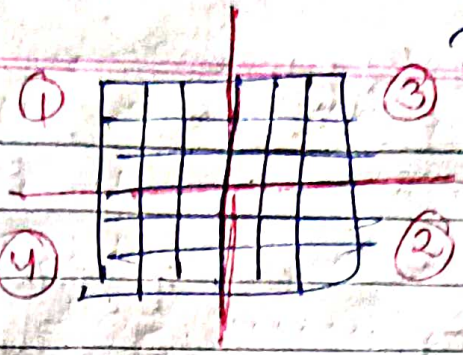
Node	Parent
S	–
B	S
H	B
G	H
E	G

Path: S → B → H → G → E

Cost = 7

Magic Square





→ order of filling

6 × 6

8	1	6	26	19	24
3	5	7	21	23	25
4	9	2	22	27	20
35	28	33	17	10	15
30	32	34	12	14	16
31	36	29	13	18	11

1 3 4 2

6 × 6

8 × 8

1	63	62	4	5	59	58	8
56	10	11	53	52	14	15	49
48	18	19	45	44	22	23	41
25	39	38	28	29	35	34	32
33	31	30	36	37	27	26	40
24	42	43	21	20	46	47	17
16	50	51	13	12	54	55	9
57	7	6	60	61	3	2	64

$$\text{sum} = \frac{n(n^2+1)}{2}$$

Algo

① First mark diagonals of all partitions too

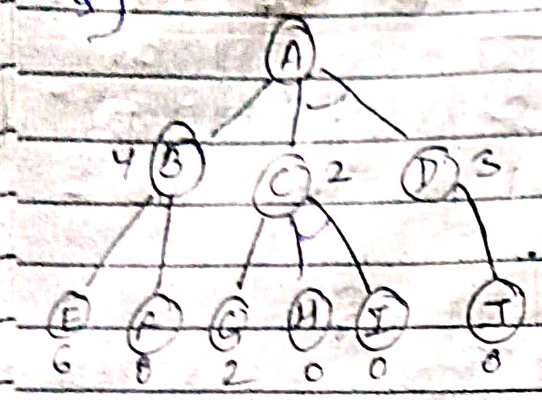
② Then fill in L to R & fill numbers in marked position & leave remaining

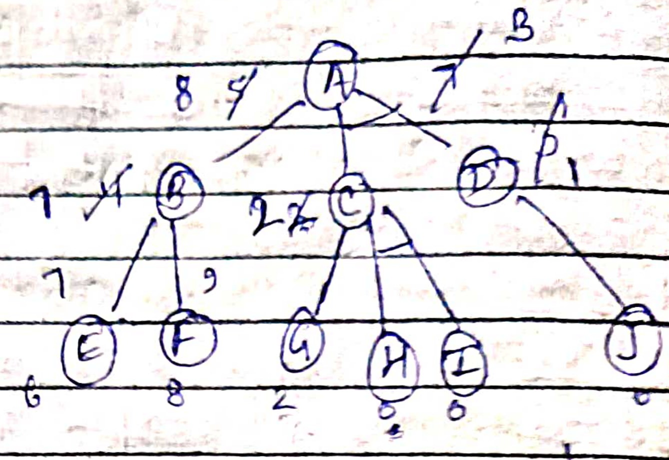
③ Then fill from bottom R to L & fill numbers in unfilled position.

09·10·2023
Monday

AO* : works on solved values

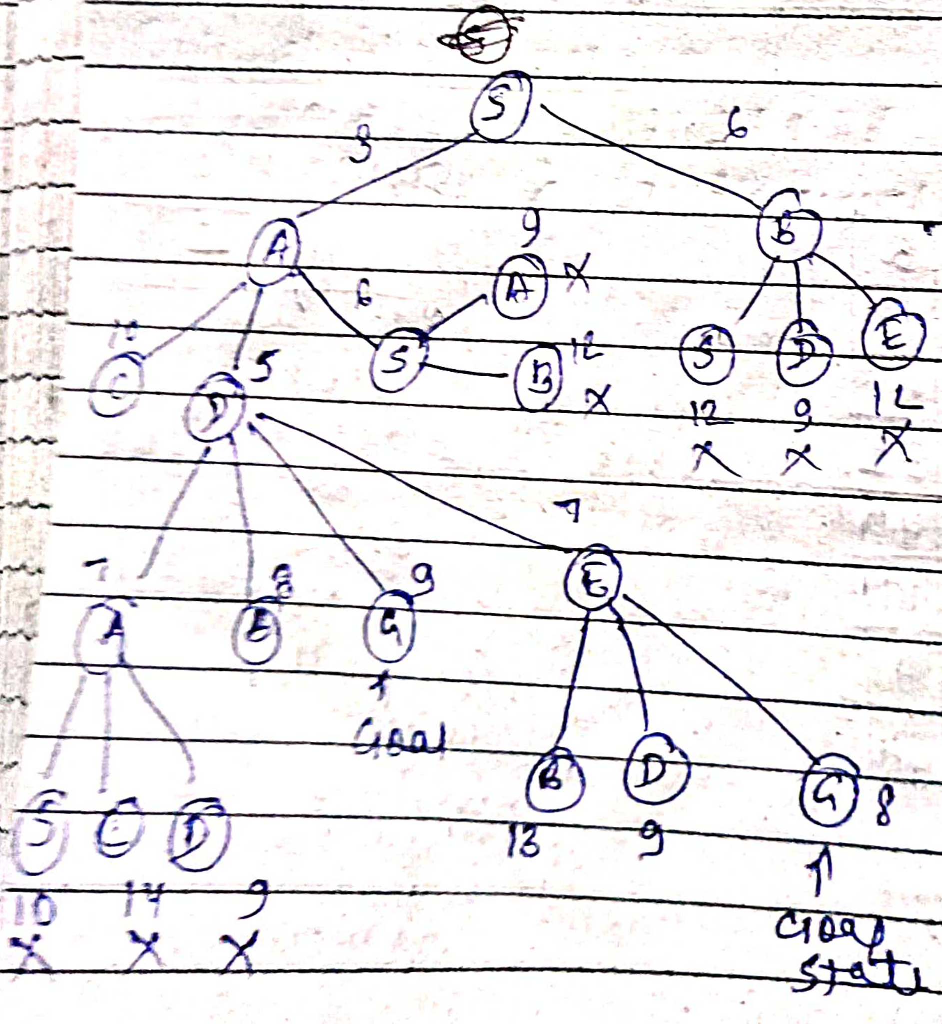
Q)

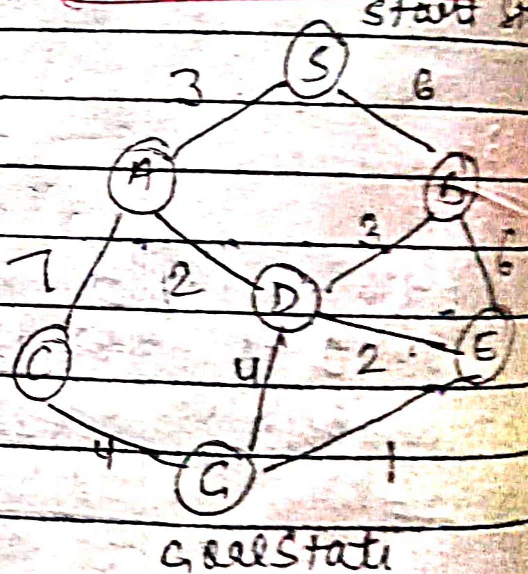




C : Solved
D : Solved

BRANCH & BOUND TECHNIQUE





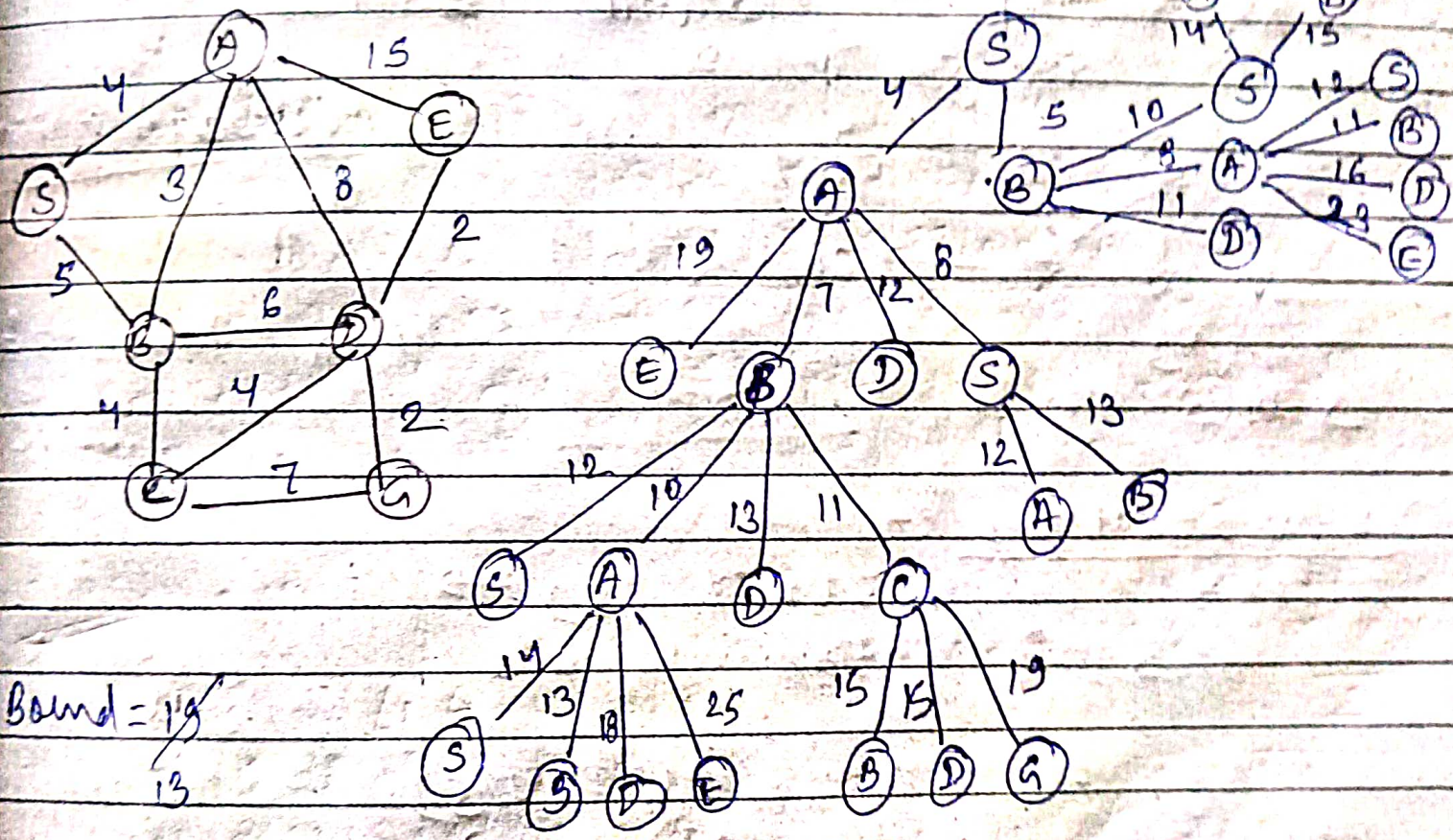
Goal पर पहुँचो तो Bound बना लो & Bound से ज्यादा वाले सब बराबर cost वाले & GS ना हो तो end that branch

Bound = 9 8 S→A→D→E→G

We ~~can~~ share repeat nodes in this approach.

A* में repeat नहीं करते nodes, Branch & Bound में कर सकते हैं।

. Space complexity of Branch & Bound > A* Algo.



Disadvantage:
Space बहुत ज्यादा लगेगी
Parallelization नहीं हो सकता।

S → B → D → G

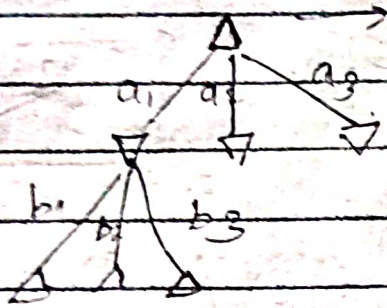
→ Game Playing
Competitive search technique

ADVERSARIAL SEARCH

Give utility values to each leaf node.

X जीता तो 1 O जीता तो -1

Tie हुआ तो 0



→ X खेलेगा

→ opp. खेलेगा

Adversarial Strategy

a_1 a_2 a_3

$a_1 b_1$ $a_2 b_2$

Optimal Strategy :-

MINIMAX = UTILITY (S) →

if S is a terminal node

recursive call

$= \max_{a \in} ACTION(S)$
if Player (S) = MAX

$= \min_{a \in} ACTION(S)$
if Player (S) = MIN

10.10.2023
Tuesday

· Create the slot & fill it

Date: Page:

Structures used for knowledge representation :-

Strong slot & filler	Weak slot & Filler
- conceptual Dependency (CD)	- Semantic Network
- Script	- frames

John ran

↓ ↓

Action Action

↓ ↓

PP ACT

Tense
f → future
P → past
Pr → present

Picture producer

Changes in physical — Change in Action

PA (Physical Adder) AA (Action Adder)

↳ termination

Actions	Example	relational → P
1. ATRANS	Give, Take,	Transfer of abstract
2. PTRANS	Go, Went, Run	Transfer of Physical location of object
3. PROPEL	Push, pull	Application of an Physical force of an object.
4. MOVE	Kick, Sit	movement of body part of owner.
5. GRASP	catch	Grasping of an object by an actor.
6. INGEST	eat	Ingestion of an object by an animal

MTrans & Speak

आइडिया/जानकारी का Transmit करना — कुछ भी बोलना

7. MTRANS — tell — Transfer of mental information

8. MBUILD — decide — Building of new information out of old.

~~9. SPEAK~~

9. SPEAK — say — Production of sound

10. ATTEND — listen — Focusing of sense organs towards the stimulus

RULES (CD)

① Describes the relationship betn the actor & the event he/she causes.

PP $\overset{p}{\Longleftrightarrow}$ ACT

→ Tense

→ CD Representation

No significance of arrow

ex. John ran

JOHN $\overset{p}{\Longleftrightarrow}$ PTRANS

② Describes the relationship betn the actor & the modifier of that actor.

PP ⟺ PA

ex. John is tall

JOHN ⟺ (height > av)

→ average

Fuzzy logic

Set define करो जो avg. कद रहे Height

avg

का

Date: ______ Page: ______

(3) Describes the relationship betwⁿ 2 PPs, one of which belongs to the set defined by others.

CD: — PP ⟺ PP

(one PP is ~~def~~ is defined as set of others)

ex: John is a Doctor

JOHN ⟺ DOCTOR

(4) Describes the relationship betwⁿ PP & an attribute that has already been predicted for it.

CD: PP ← PA

(1 is predicted but 2 one is not)

ex: A nice boy.

BOY ← NICE

(5) Describes the relationship betwⁿ 2 PPs, one of which provides a particular (kind) type of information about others.

PP ⇐ PP

ex: John's dog

John ⇐ (possessed by) Dog

(7) ~~(6)~~ Describes the relationship betwⁿ an action & the source & recepient of that (of that) action.

ACT ← R → PP, ← PP

ex: John took the book from Mary.

JOHN ⟺ ATRANS ← R → JOHN, ← Mary

ATRANS ←o— book

1 6 7

16/10/2023 Monday

Use of Itf ← John presents many to gives book to other person.

Date: Page:

(6) Describes the ~~rule~~ relationship betⁿ an Action & ~~that~~ PP that is the object of that action.

ACT ←° PP

John pulled the cart

JOHN ←ᴬ PROPEL ←° CART

6

(3) Describes the relation betⁿ an ACT & the instrument with which is informed

ACT ←ᴵ ⇕

John ate icecream with a spoon

1 6 3

JOHN ⇐⇒ INGEST ←ᴵ DO → अब पत्र पूरा हो।

↑ᴼ ↑ᴼ

Icecream Spoon

6 6

(9) Describes the relationship betⁿ an act & its physical src. & dest.

ACT ←ᴰ → PP

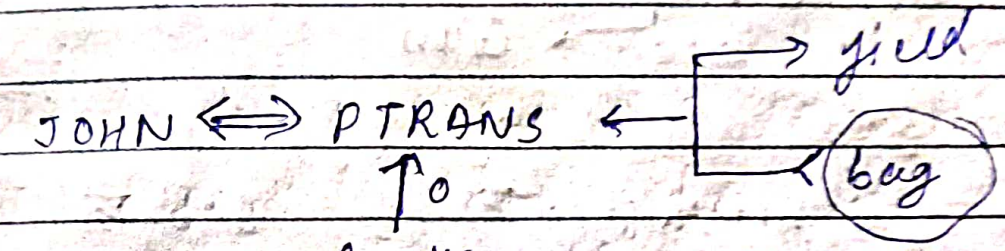
→ PP

⑦ & ⑨ में R & D का अंतर |

ATRANS — PTRANS हमेशा

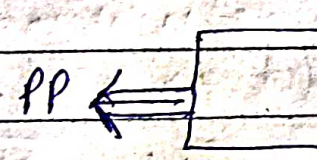
John fertilized the field.

fertilizer पहले bag में था फिर field में



use करने के बाद bag empty हो जाता है

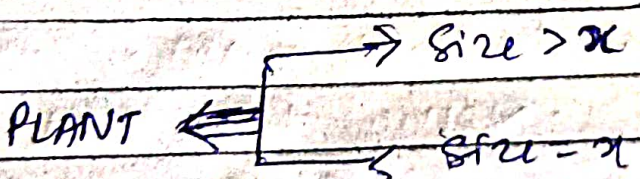
⑩ Represents the relationship between a PP & the state in which it is started and the other in which it is ended.



modification हो रहा है PP में |

The plant grew

PP

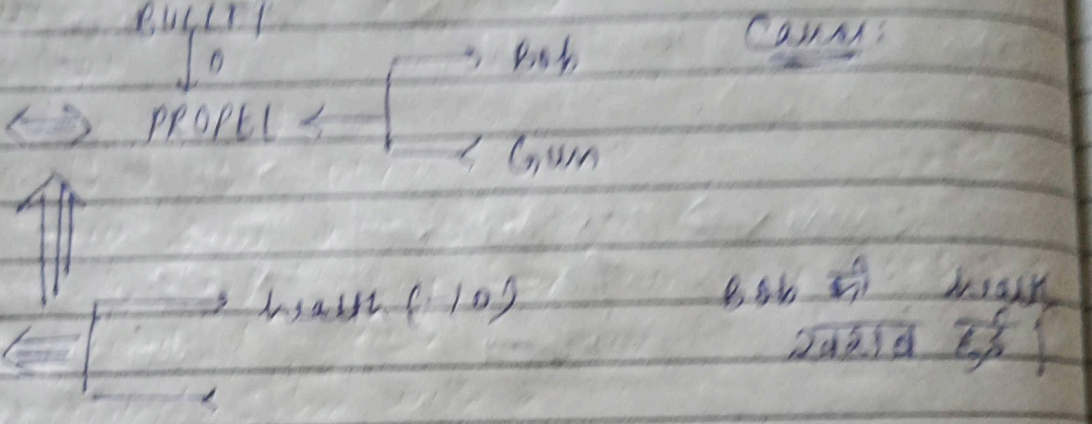


⑪ Describes the relation between one conceptualization & another that causes it.

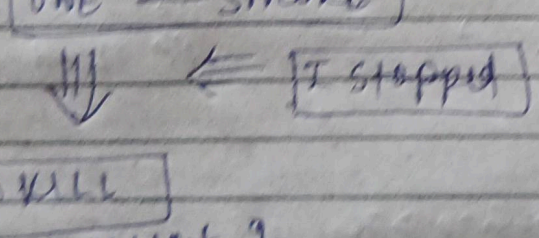
एक concept को बदले दूसरा concept हुआ।

Bill shot Bob

Causal Reason — Health of Bob is not good.

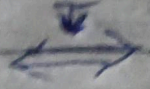


Since smoking can kill you, I stopped



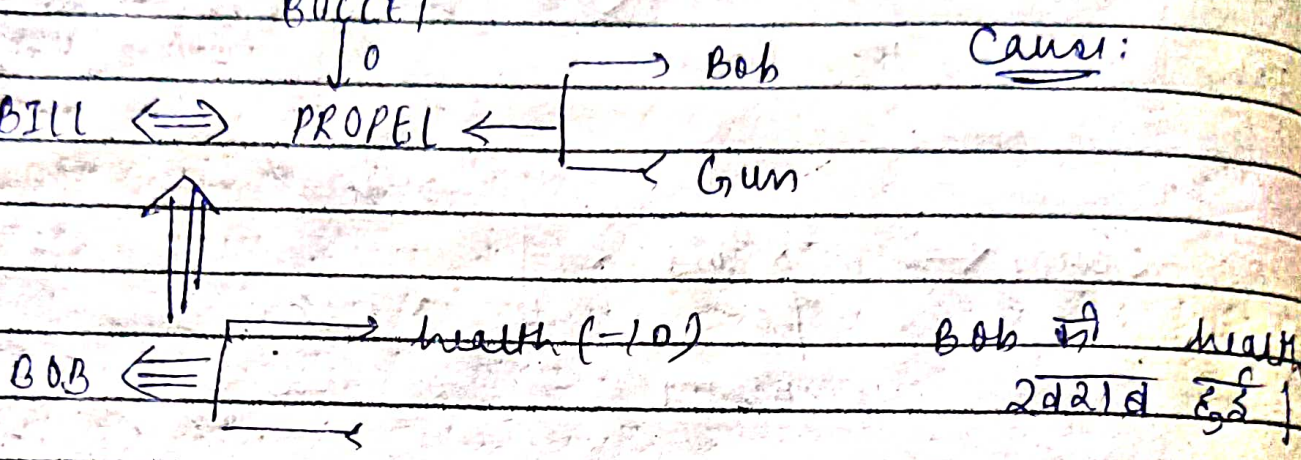
(22) Describe the relation b/t a conceptualization & the time at which the event it describes occurs

John ran yesterday

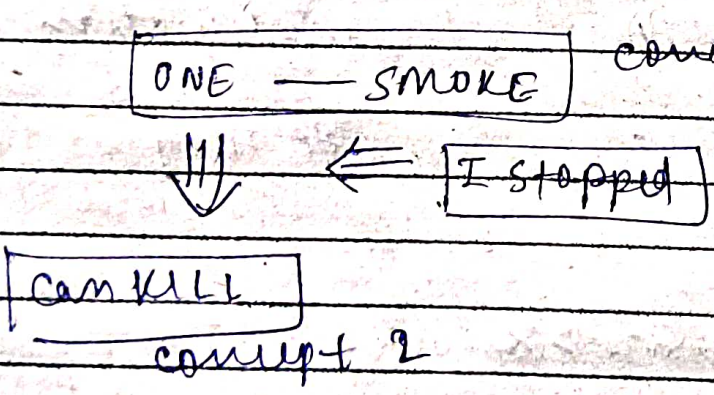


Bill shot Bob

Causal Reason — Health of Bob is not good.

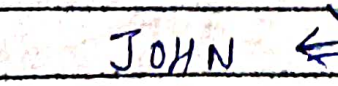


Since smoking can kill you, I stopped



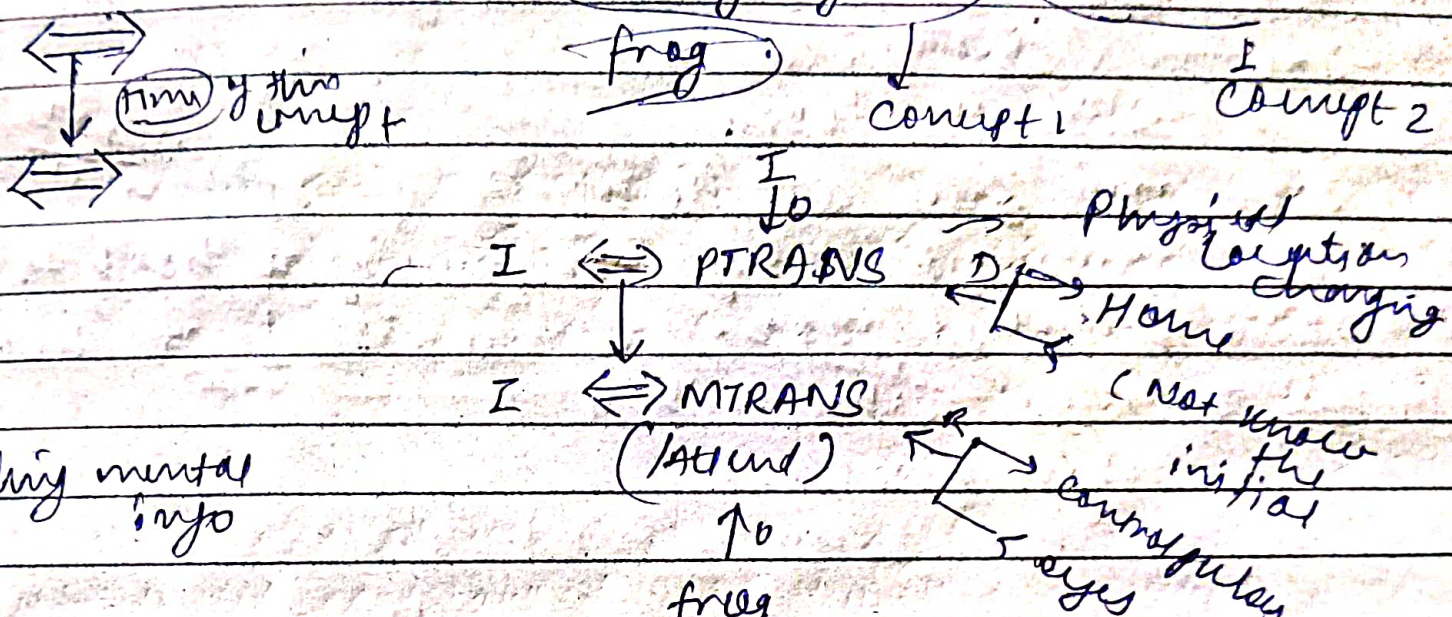
(12) Describe the relation between a conceptualization & the time at which the event it describes occurs.

John ran yesterday

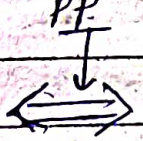


(13) Describes the relationship bet" 1 conceptualization & another that is the time of the first

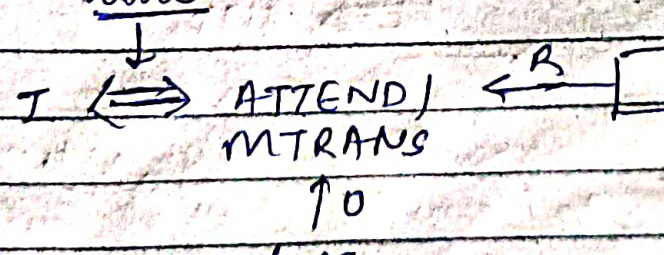
(Time of A concept E)



(14) Describes the relationship with conceptualization & the place at which it occurs



I heard a frog in the woods.



Date Page

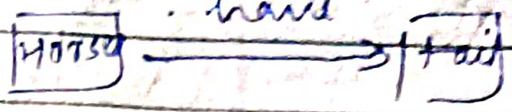
(1) John ate the egg
(2) John bigged many for a pencil
(3) I gave the man a book
(4) Bird flew
(5) Joe is a student.
(6) Joe pushed the door.
(7) Joe gave Sue a flower
(8) John promised many for giving a book to Bi
(9) John punched Joe.
(10) Bill threatened John with a broken nose.

17.10.2023
Tuesday

→ Weak Representation

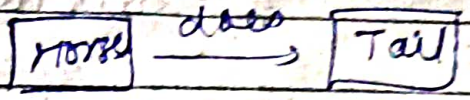
Semantic Networks

Horses have tail.



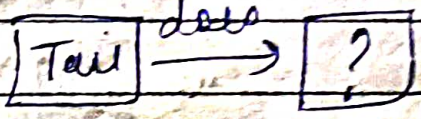
(Name is Tom
Born)

Does horse have tail ?

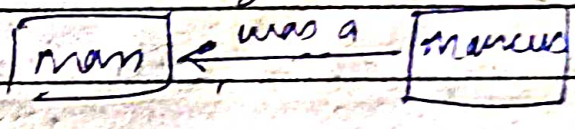


~~Sol~~ Try to fit in Semantic N/w

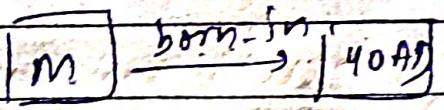
What does horse have?



Marcus was a man. Predicate logic man(marcus)



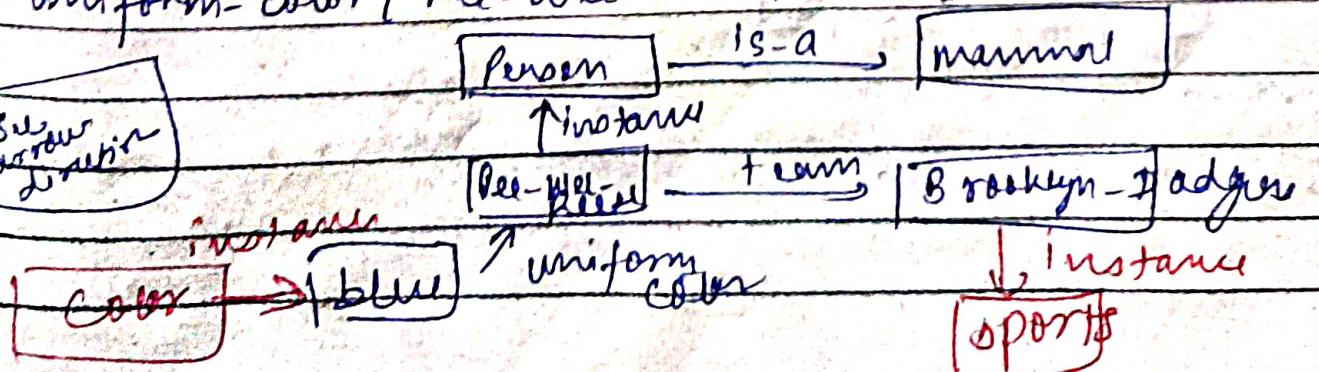
born_in(M, 40)



predicate logic : born-in(M, 40)

Q) Consider the following statements

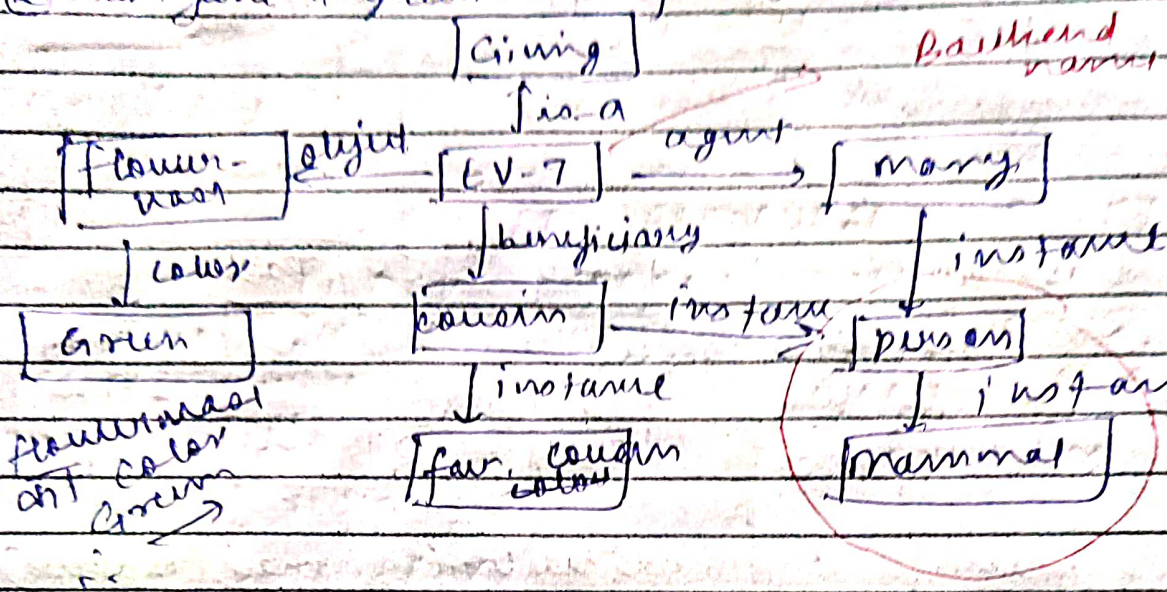
① is_a (Person, Mammal)
instance (Pee-Wee-Reese, Person)
team (Pee-Wee-Reese, Brooklyn-Dodgers)
uniform-color (Pee-Wee-Reese, blue)



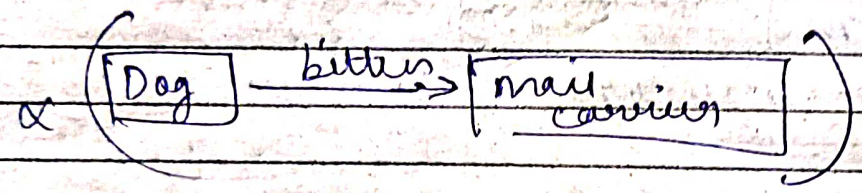
Blue bird
vertebrate

Date: Page:

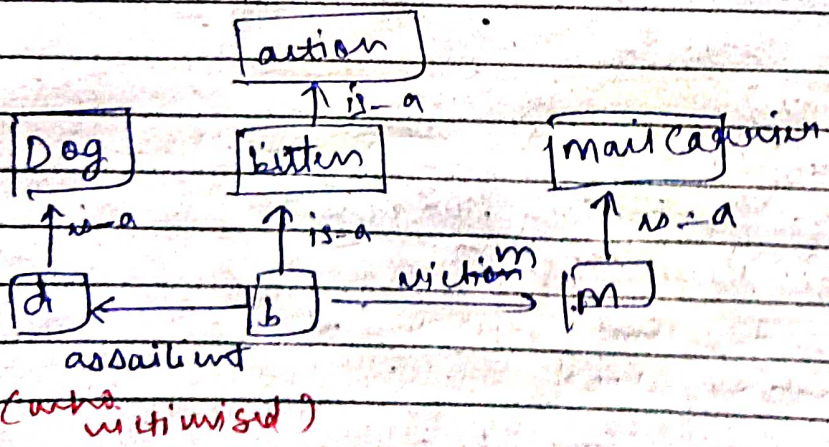
② Mary gave a green-color flower to her favourite c



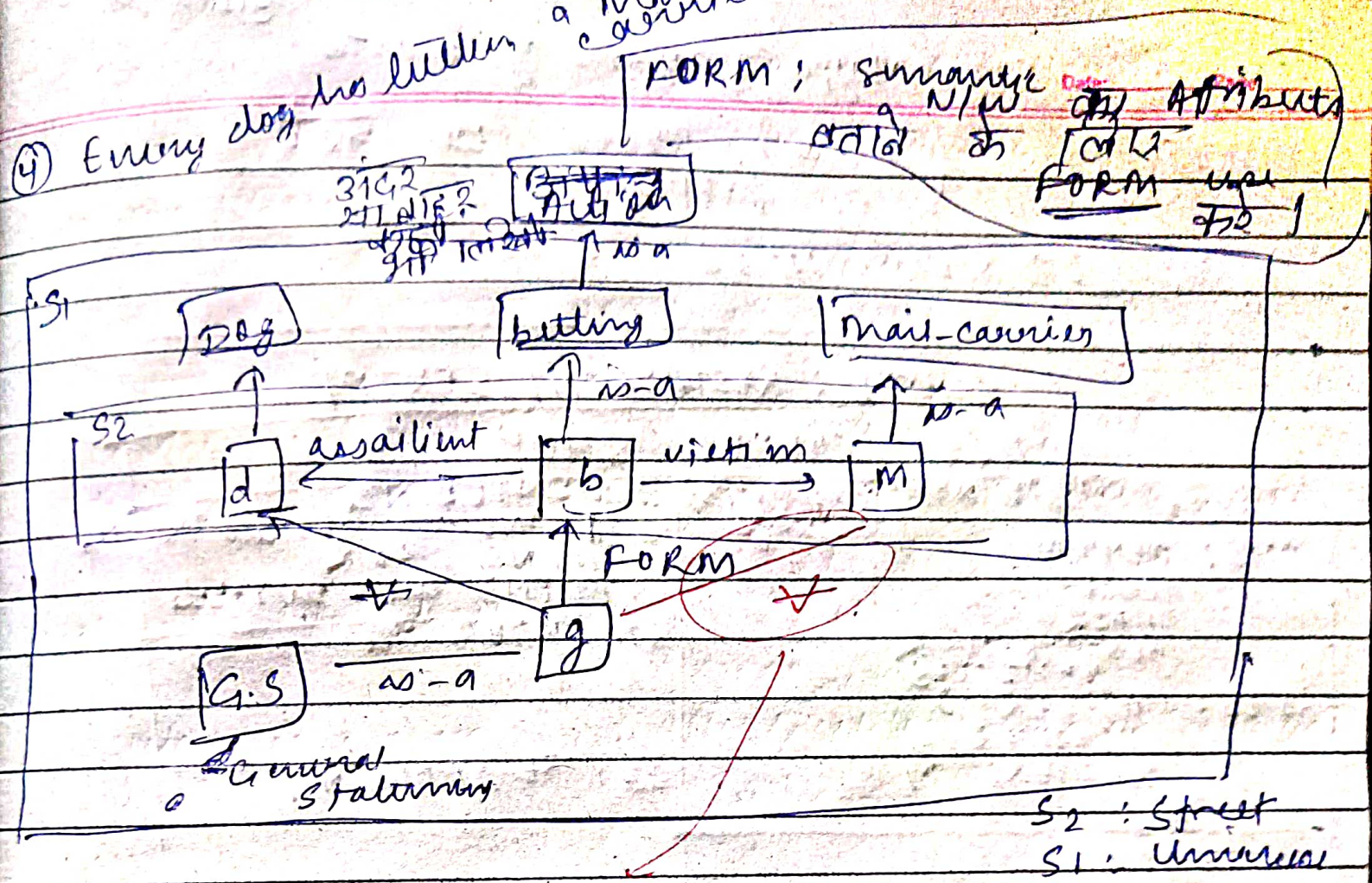
③ The dog has bitten a mail carrier.



- This is for gene
dog & mail ca
" But the question
specific for dog
& mail ca



(4) Every dog has bitten a mail carrier



(5) Every dog has bitten every mail carrier

Same ques. Every batter hit every ball
Every batter hit

Strong Slot filler representation

Ye perfect rectangle होना चाहिए

SCRIPT ← Stereotype sequence of events (Atrans, Ptrans use करके props etc)

↳ Using CD Representation

SCRIPT : RESTAURANT SCRIPT	
TRACK : COFFEE SHOP	
// Living (Roles) & Non Living (props) things	S1: ENTERING ~~C PTRANS C's vehicle to PARKING~~ C PTRANS C to R ~~C ATTEND eye's on Table~~
PROPS: MENU-M TABLE CHAIR RESTAURANT-R VEHICLES	S2: ORDERING
ROLES: CASHIER-O CUSTOMER-C WAITER-W	S3: RE-ORDERING
ENTRY CONDITIONS: C should have money R should be open	S4: BILL PAYMENT
RESULT: C has less money O has more money C is pleased	S5: EXITING

→ Script से पहले की cond^n

→ Script के बाद की cond^n

→ Perfect Rectangle

C MBUILD "where to sit"

C PTRANS "to that table"

कम से कम 5-7 scenes & in each scene atleast 5 to 7 statements

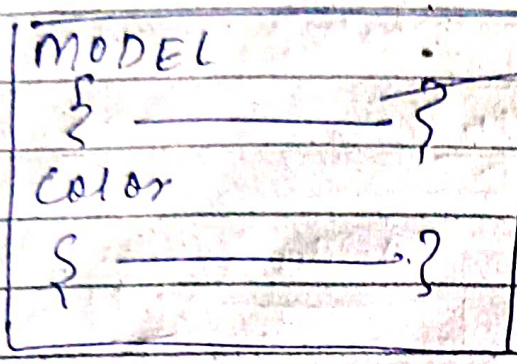
CD ← → Simple english

Recall ~~Page~~ Persistence

→ आपके mind

Date _____ Page _____

FRAMES



→ आपके दिये गए में कौनसा आएगा.

?

* ~~Inheritance~~ है या नहीं है इसमें।