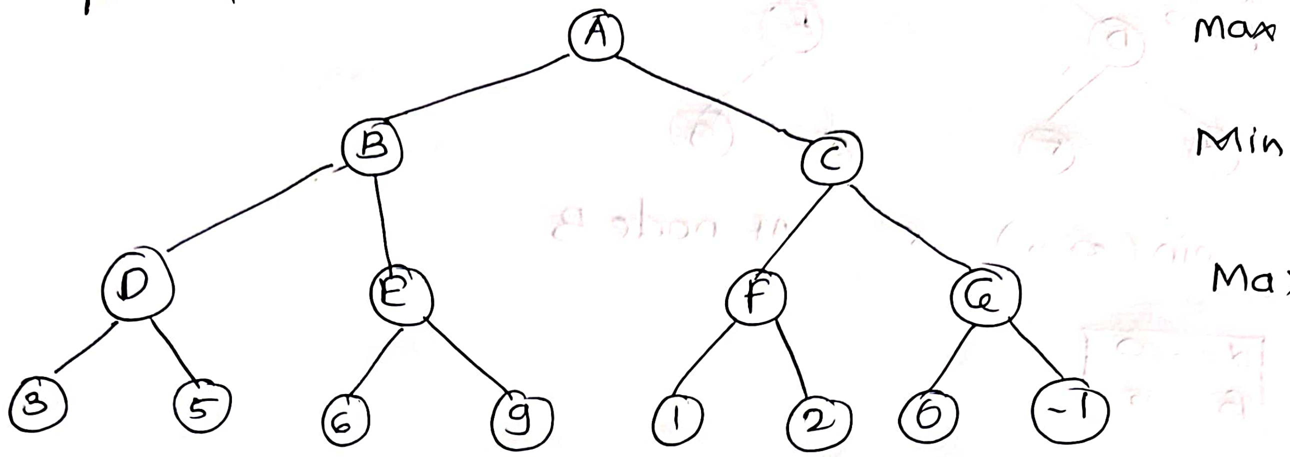
Que 1 Perform alpha-beta pruning based min-max algorithm on the following graph. Show alpha-beta for each node at the end, and also show which node will be pruned.



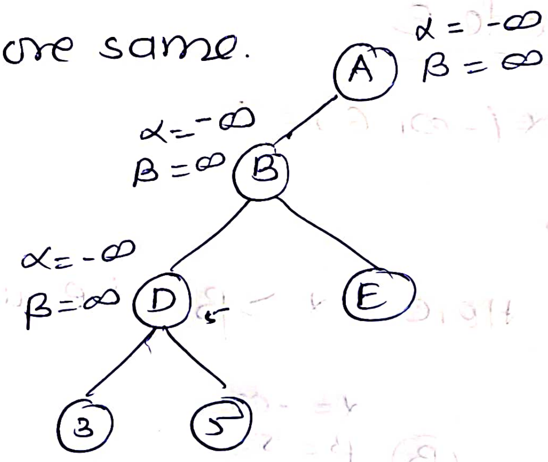
⇒ Initialize α & β

α → represents the best value that the maximizing player (MAX) can guarantee

β → represents the best value that the minimizing player (MIN) can guarantee

Intially at <u>Node A</u> $\alpha = -\infty$ (for Max)
$\beta = \infty$ (for MIN)

at node B & D will more same.

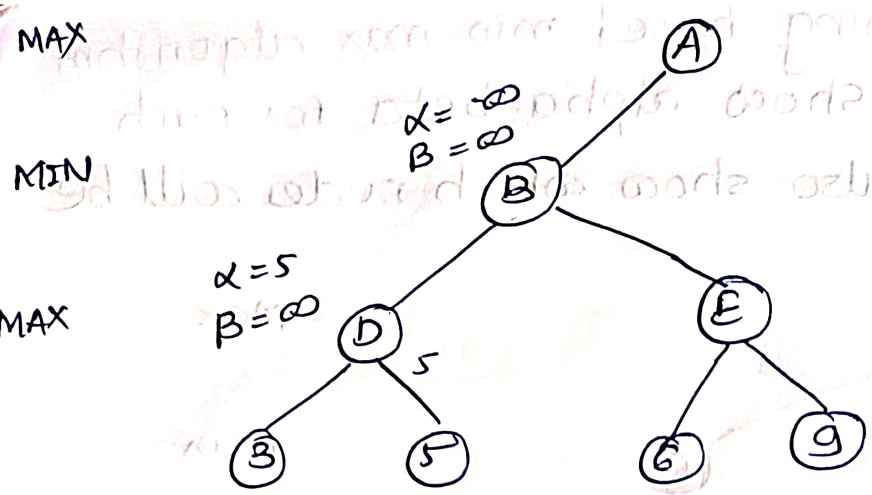


∴ <u>At node D</u> (MAX layer)

$\alpha = \max(-\infty, 3) = 3$ updated $\alpha = 3$
$\beta = \infty$

again
$\alpha = \max(3, 5) = 5$ updated $\alpha = 5$
$\beta = \infty$.

<u>At node B</u> (MIN Layer)
B will be updated



$\beta = \min(\infty, 5) = 5$ At node B.

$\therefore$ $\boxed{\alpha = -\infty \quad \beta = 5}$

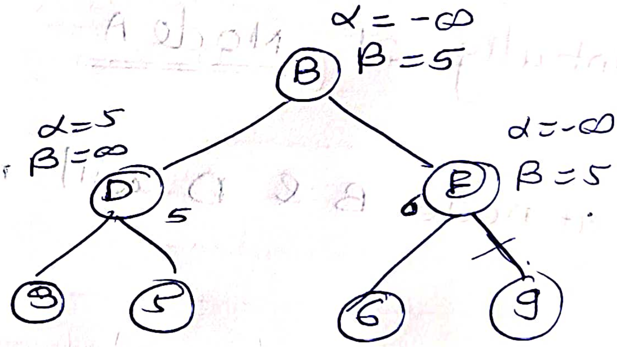
~~At node A (MAX node)~~

~~α will be updated~~ ~~$\alpha = -\infty$~~
~~$\beta = \infty$~~

~~Now $\alpha = \max(-\infty,$~~

Now will check right side of node B

Now at node E

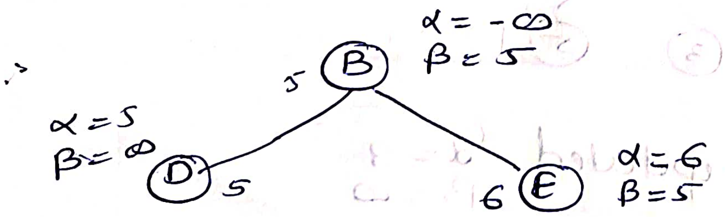


$\alpha = MAX(-\infty, 6) = 6$

$\therefore$ $\alpha = 6$
$\beta = 5$

Here $\alpha \geq \beta$ i.e will prune the 9 node

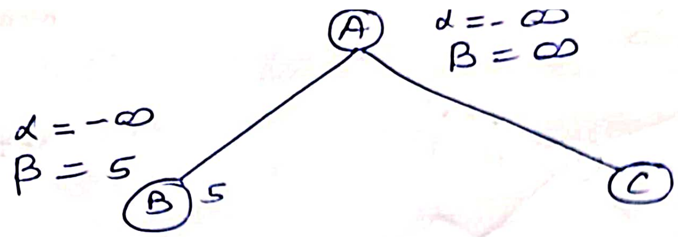
$\therefore$



At node B (MIN layer)

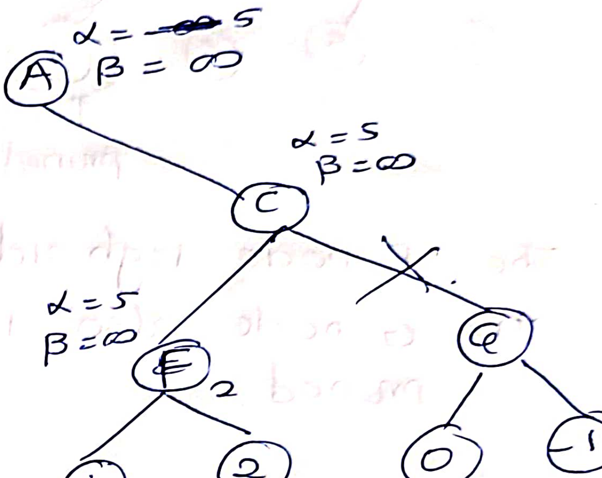
β will be updated $\beta = MIN(5, 6) = 5$

$\therefore$ $\boxed{\alpha = -\infty \quad \beta = 5}$



At <u>Note A</u> (MAX Layer) $\alpha = MAX(-\infty, 5) = 5$

$\therefore$ $\boxed{\alpha = 5 \quad \beta = \infty}$



At <u>Node F</u> (MAX)

$\alpha = MAX(5, 1) = 5$

$= MAX(5, 2) = 5$

$\therefore$ $\left.\begin{matrix} \alpha = 5 \\ \beta = \infty \end{matrix}\right\}$ No change.

<u>node value 2</u>

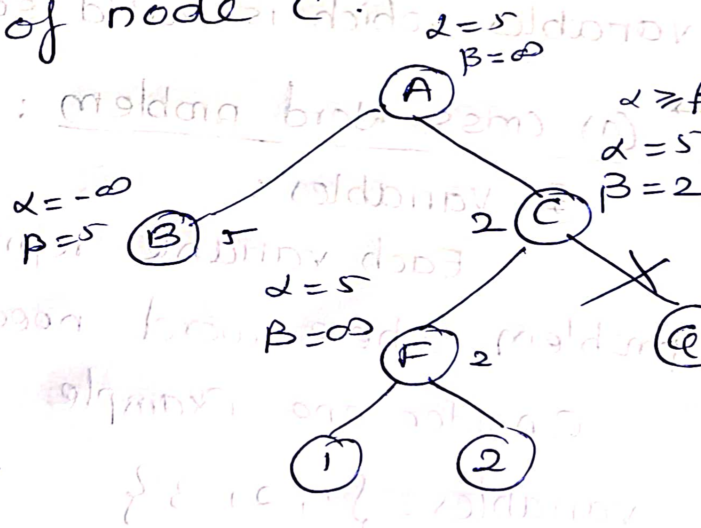
At <u>node C</u> (MIN)

β will update $\beta = MIN(\infty, 2) = 2$

$\therefore$ $\alpha = 5$ $\quad$ $\alpha \geq \beta$ break

$\beta = 2$

will be pruned right side of node C.



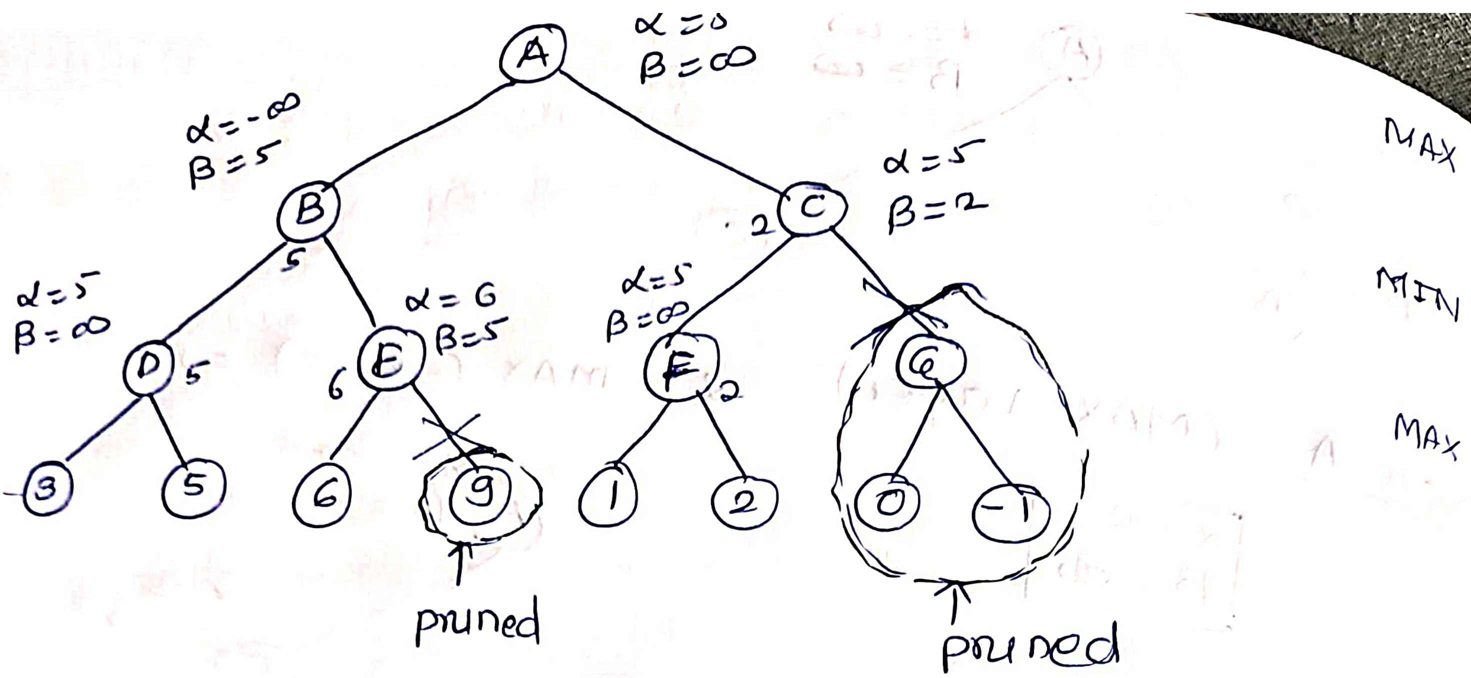
$\therefore$ At <u>node C</u> $\alpha = 5$

$\beta = 2$

node value = 2

At <u>node A</u> $\alpha = MAX(5, 2) = 5$

$\therefore$ $\boxed{\alpha = 5 \quad \beta = \infty}$

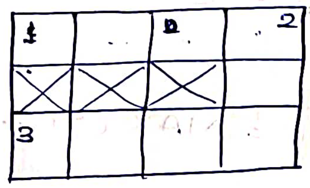


The E node righ side will be pruned
The G node also i.e right side of c node will be pruned.

Que.8 Formulate the following problems as CSP search.
(a) cross word problem (c) Airlines Gate scheduling problem
(b) 4-queen problem (d) Map coloring problem.
For each of these problem state: variables, domain, constrain optimization criteria (if any) and give one assignment to variables which is valid solution

→ (a) Cross Word problem :

① Variables :
Each variable represents a slot in the cross word problem where word needs to be placed.
Consider one Example :
vanables = {1, 2, 3}.



② Domain :
Now domain for cross word problem is set of words that can fit in the corresponding slot
In above Example Domain = { Frog, word, dog }

③ Constraints :
- The word length must match with the slot length
- Word must share common letters at intersection.
- No word can be repeated.

④ optimization criteria : there is no such optimization criteria for cross word problem.

⑤ Valid Solution :

①W	o	r	d②
			o
③F	r	o	g

1. Word
2. dog
3. frog

} solution.

(b) 4. Queen problem :

① Variables : → Each variable represent a queen's position on the chessboard.

Example : $\{Q_1, Q_2, Q_3, Q_4\}$ where Q_i → row position of the queen in ith column.

② Domain :- Each set of possible rows for each queen.

Domain = {1, 2, 3, 4}

③ Constraint : → No two queens can be in the same row.
- No two queens can be in the same column.
- No two queens
- This is implicity satisfied by having one variable per column.

Also No two queens ca be on the same diagonal.

$|Q_i - Q_j| \neq |i - j|$ for $i \neq j$.

④ optimization criteria : No optimization, purely constraint based

⑤ Valid solution

	a	b	c	d
1		Q_1		
2				Q_2
3	Q_3			
4			Q_4	

$Q_1 = (1, b)$	$Q_2 = (2, d)$	$Q_3 = (3, a)$	$Q_4 = (4, c)$

This is valid solution for 4 queen problem.

(c) Airline Gate scheduling problem:

① **Variables** : Each variable represent flight that need to assigned to gate. like, $F_1, F_2, \ldots F_n$

variables = $\{F_1, F_2, F_3, F_4\}$

F_1 : 6 AM
F_2 : 2 PM
F_3 : 6:05 AM
F_4 : 10 AM

② Domain : set of gates available

Domain : $\{G_1, G_2\}$

③ Constraints :

- No two flight can be assigned to same gate at the same time or with less time gap.
- Certain gates might be reserved for certain types of flights (International vs domestic).
- Minimum turnaround time between consecutive flights at same gate must be respected.

④ Optimization criteria :

- Minimize the total delay of flights
- Maximize the efficient use of gates
- Minimize gate changes for passengers transferring between flights

⑤ valid solution :

we have 2 gates G_1, G_2

F_1 : 6 AM
F_2 : 2 PM
F_3 : 6:05
F_4 = 10 AM

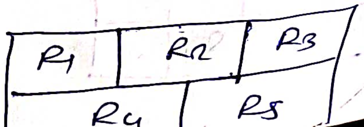
→ As the difference between F_1, F_2 & F_4 is large so we can assign to same gate.

$\therefore$ $\boxed{F_1 = G_1, \ F_2 = G_1, \ F_3 = G_2, \ F_4 = G_1}$

) Map coloring Problem:

① Variables : Variables here represent the region or country that needs to be colored.

take example : $\{R_1, R_2, R_3, R_4, R_5\}$



② Domain : Here the set of colors = { Red, Blue, Green }

③ Constraint :
Adjacent Regions must have different colors (i.e no two neighbours can share same color).

④ optimising criteria :
The optimization criteria is to minimize the number of colors used.

⑤ Valid solution :
we have 5 Regions R_1 R_2 R_4 are neighbours
∴ R_1 = Red , R_2 = Green , R_4 = Blue
R_1 = R_5 } color as those are not neighbours
& R_3 = R_4 }
∴ R_5 = ~~Blue~~ Red & R_3 = Blue.
∴ valid solution is R_1 = Red , R_2 = Green, R_4 = Blue, R_3 = Blue , R_5 = Red.

Que.5. Apply BFS, DFS, UCS, Best first & A* search on the following city map. You are starting at Arad and your goal is to reach Bucharest.

⟹ Given Graph with heuristic distance & the actual distance g(n).
Let's assign the unique A-Z letter for each city
The straight line / heuristic distance given as

Arad (A) = 360
Bucharest (B) = 0
Craiova (C) = 160
Dobreta (D) = 242
Eforie (E) = 161
Fagaras (F) = 178
Giurgiu (G) = 77
Hirsova (H) = 151
Iasi (I) = 226
Lugoj (L) = 244
Mehadia (M) = 241
Neamt (N) = 234
Oradea (O) = 380
Pitesti (P) = 98
Rimnicu Vilcea (RV) = 193
Sibiu (S) = 253
(T) Timisoara = (329)
(U) Urziceni = (80)
Vaslui (V) = 199
Zerind (Z) = 374