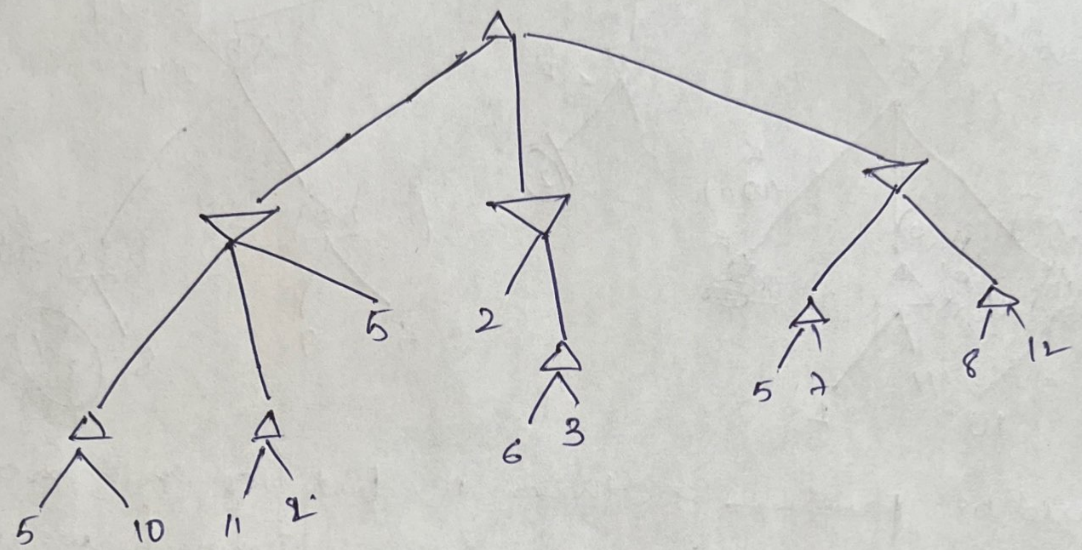
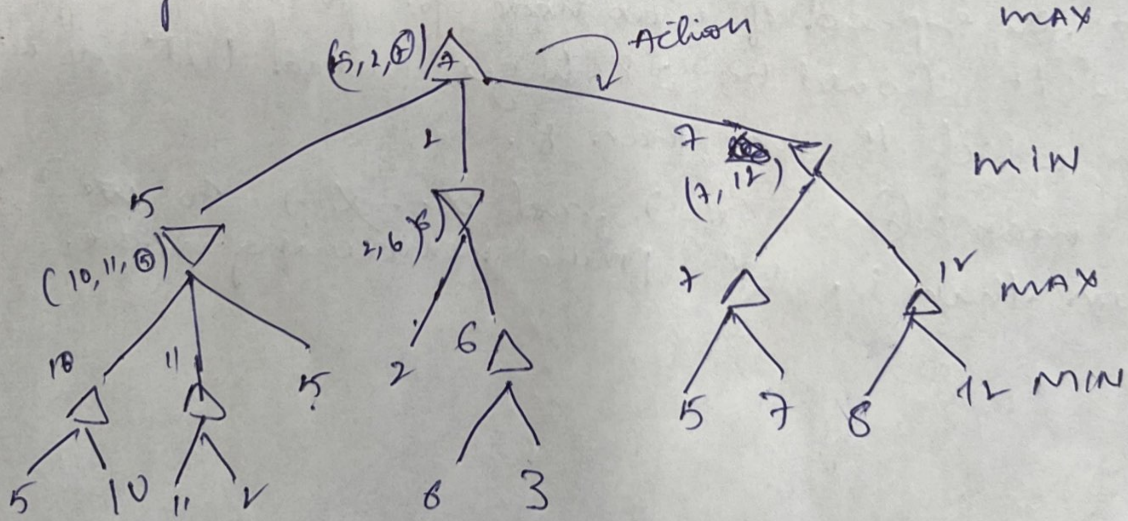
Sai Parthish Mandumula
1002022847

Task 1:-

In the game search tree of the given fig 1: we are given minmax algorithm

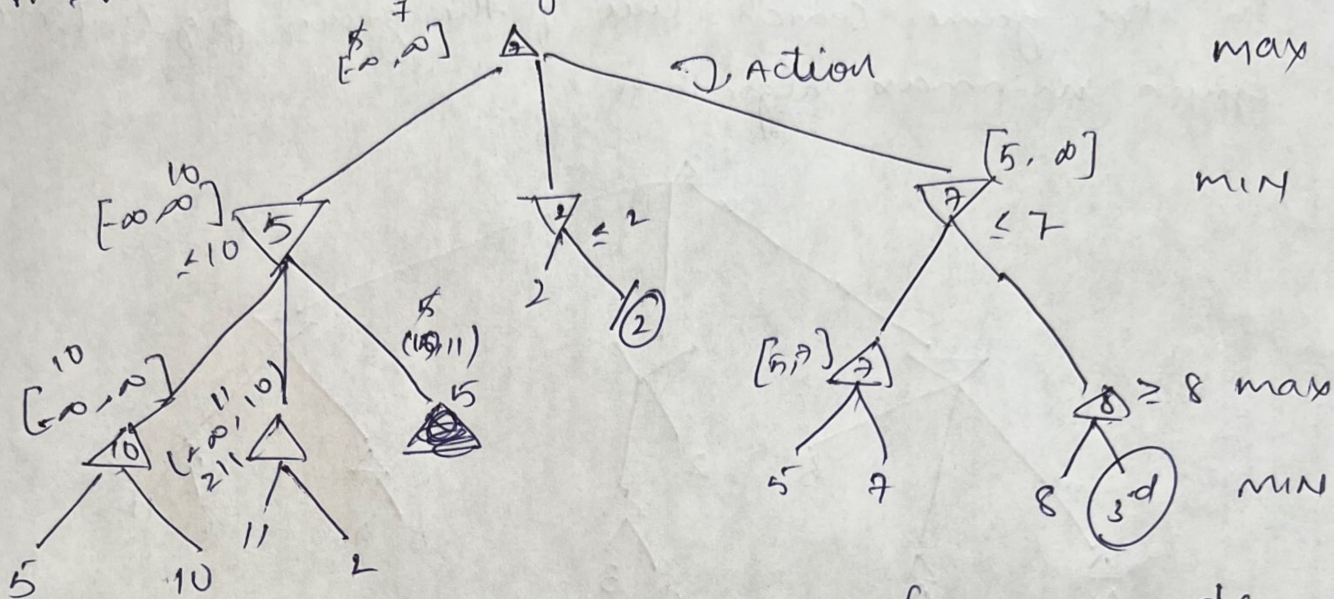


By using minmax search using Depth-first search strategy



By using minmax search we got answer as 7 (highest Achievable) against optimal opponent.

(5) By using α-β. search. (left right order) we ~~go~~ do alpha → best highest-value → (−∞) and Beta → best lowest-value (+∞). we get same answer as past, i.e 7.



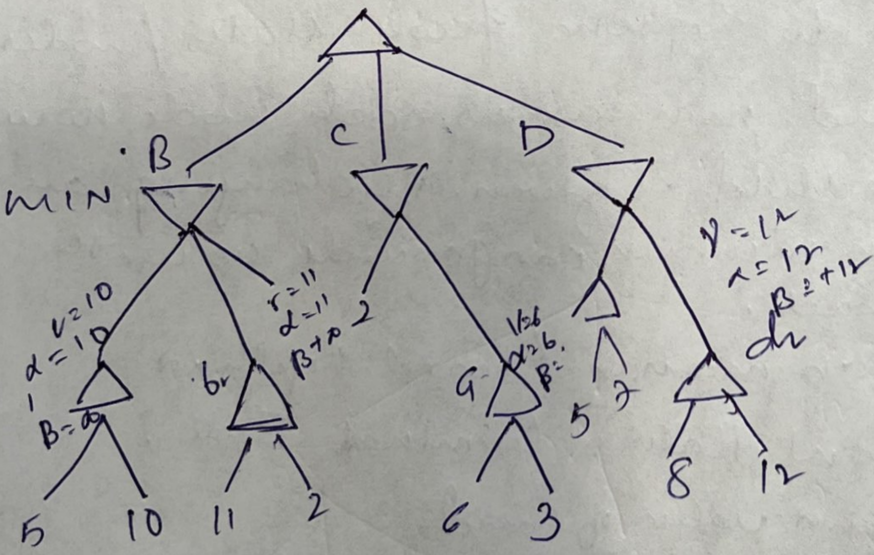
It should be ~~...~~ than 11, but for min node it will check to go remaining nodes get pruned. we get 2 but we already on expand min node, so, it should be ≤ 2. we got 7 and 8 have 5. On expand of max node for far right and then it should be ≥ 8 to get picked but as we had 7 then which is less than 8.

max = 7 v = 7. and α = ~~∞~~ 7 so we 7 which is less than 8. we prune remaining node.

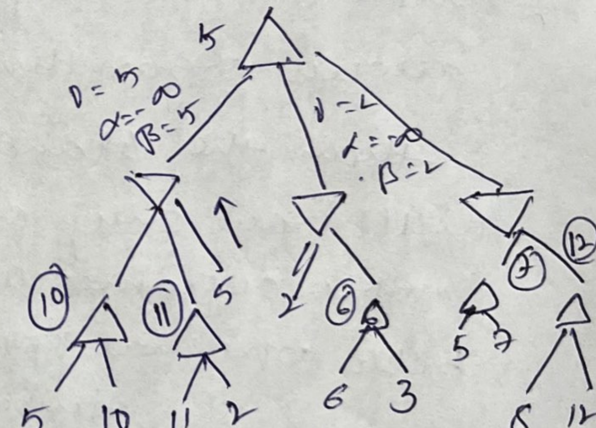
Scanned with CamScanner

(c) given additional knowledge of max – utility 12 & min. utility. we are Suppose given additional knowledge about game. So, the maximum utility Value 12 and no, outcome greater than 12 and minum utility value is 2 and outcome lower than 2 then on finding value 12 as its one of child & next step Exploring node with its child value of 2

(i) max



(ii) MAX



min child returned & stop explore.

Sai Parthish Mandumula
1002022847

Task=2

The Deep Green Move (s) will give the Exact that will compete on a two player determine game of perfect information. Also they haved given our opponent is a Super Computer. If we give State as input to deep green (s). we can modifify minmax Algorithm by changing min value and function MIN value (state) and if terminal test (state) ~~and~~ returning utility value if min value (utility state) and max value (utility state) max value (Deep green-moves) and this modification allows algorithm to Explore fewer States / nodes than the standard minmax as each node now will have only one child as given. By changing all possible outcomes win minmax can provide us with better optimal approach.

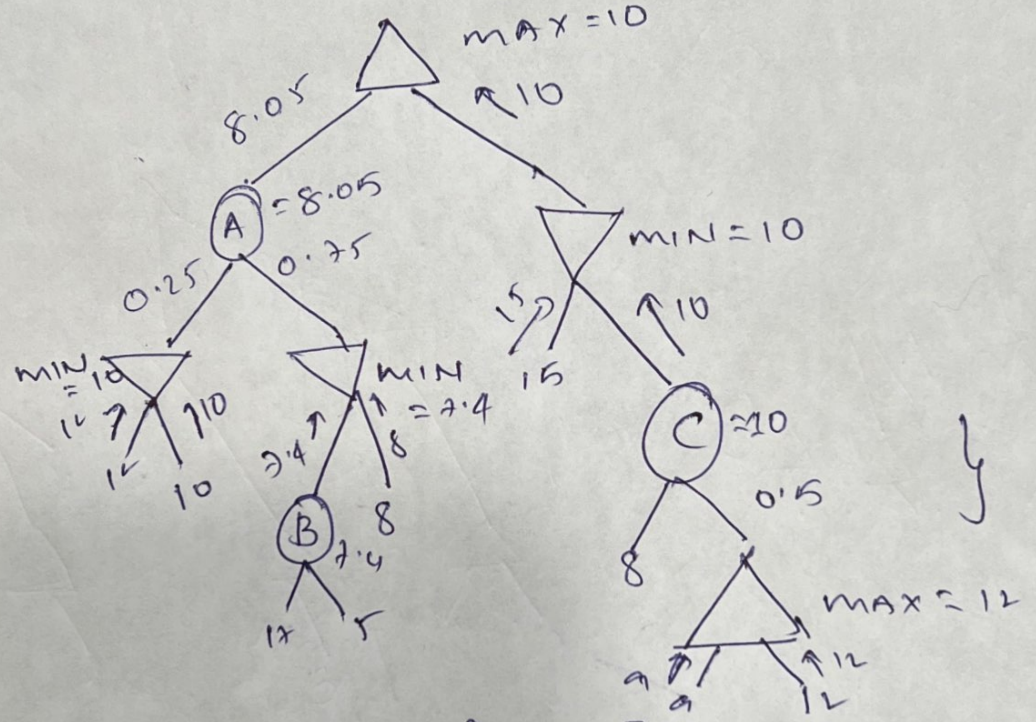
function minmax (board, depth, n)
if current State is terminal State:
return value of board
else maxplayer
Best value = Infinity.
for each move in board:
value = minmax (board, depth+1, false)
bestval = max (bestval, value)
return best value

Sai Parthish Nandurmule
1002022847

else
Best val = + infinity

for each move in board:
value = minmax (board, depth +1, true)
best value = min (Best value, value)
return Best value.

Task 3



on B = Avg [(17 × 0.2) + (5 × 0.8)]
(3.4 + 4) = 7.4

and Avg ((8 × 0.5) + (12 × 0.5))
(4 + 6) = 10

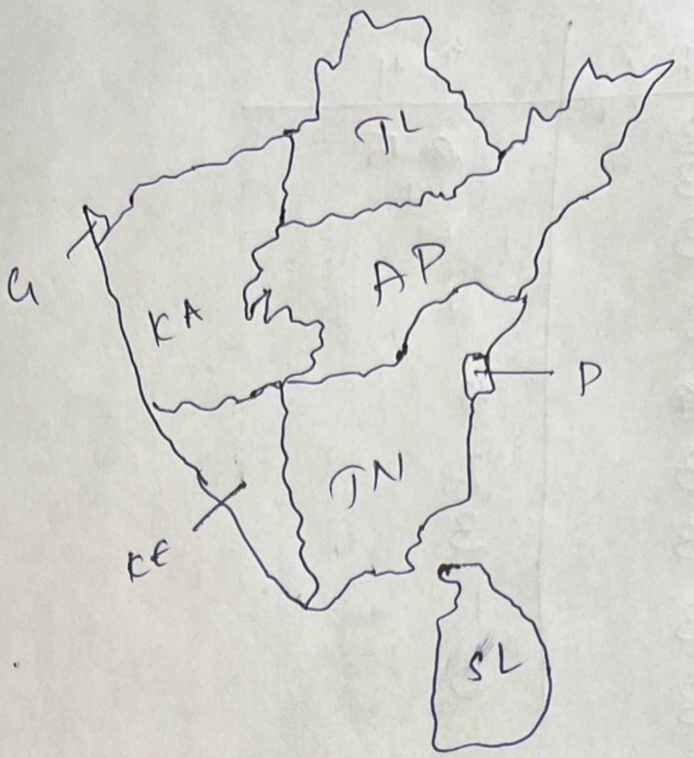
and A = Avg ((10 × 0.25) + (7.4 × 0.75)) (2.5 + 5.55)
= 8.05

The Expectminimax the outcome = 10

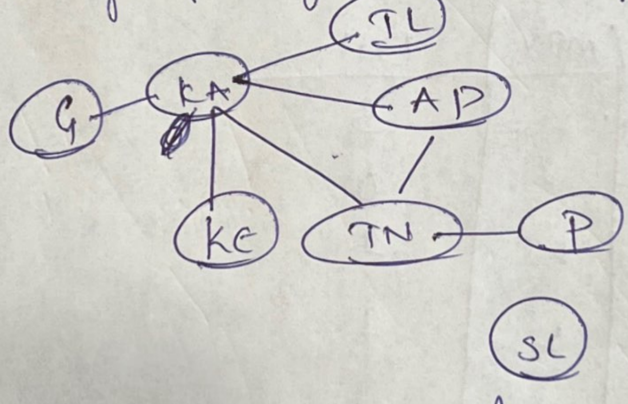
Scanned with CamScanner

Sai Parthish
1002022847

Task: 4



ⓐ Constraint Graph for this Problem.
The graph of this map



and now there is Separate node SL which is

the nodes are variable and the arc constraints.

ⓑ Assuming you are using Backtracking Search to solve this problem and that you are using MRV with degree heuristic to select the variable, and all variable will be selected at each level of the search tree

(i)

	MRV	DH
KA	3	~~4~~ 5
TN	3	4
Kt	3	2
G	3	1
~~KB~~	~~3~~	~~6~~
TL	3	2
AP	3	3
P	3	1
SL	3	0

Now, let us select 'KA' min-value with high degree heuristic.

(ii)

	MRV	DH
TN	2	3
G	2	0
KB	2	1
TL	2	1
AP	2	2
P	3	1
SL	3	0

Now, on the second we select 'TN' min-value with high degree heuristic.

Sai Parthish
1002022847

iii)

	MRV	DH
G	2	0
KE	1	0
TL	2	1
AP	1	1
P	2	0
SL	3	0

Let us ~~to~~ select 'AP' min-value with high heuristic then

	MRV	DH
G	2	0
KE	1	0
TL	1	0
P	2	0
SL	3	0

Select KE with min-value and heuristic
Select TL with min-value of DH high
and then we select G with min-value
and P followed by SL. So, the first
Order is KA ~~B~~TN AP KE TL G P SL

(c)

KA	TN	AP	KE	TL	G	P	SL
R Red	G B ↓ Green	G B ↓ Blue	G B ↓ Blue	G B ↓ Blue	R G B ↓ green	R G B ↓ Red	R G B ↓ green/blue

Assume ~~you~~ that we assign the color 'Red' to the first variable selected in part b. and above are the following steps that involved in checking the remaining legal values for all other variable using Arc Consistency then we take the first order from (b) and assigning red and giving colors that no two have repeated colors beside using arc consistency.

(d) Yes, we can use structure of the problems to make solving it more efficient for example, ~~so~~ let us take KA - P connected to standard constraints on specific problem, so then which can solve the AP & KE set. ~~so~~ Be then by elimating loops in graph it becomes a tree structure. 'SL' is solved on its own because it is independent i.e not connected to other nodes.

Sai Parthish
Mandumula
1002022843.

(e)

One valid solution is

KA = Red

TN = Green

AP = Blue

TL = green

P = Red

KE = Blue

G = Green

SL = Blue.