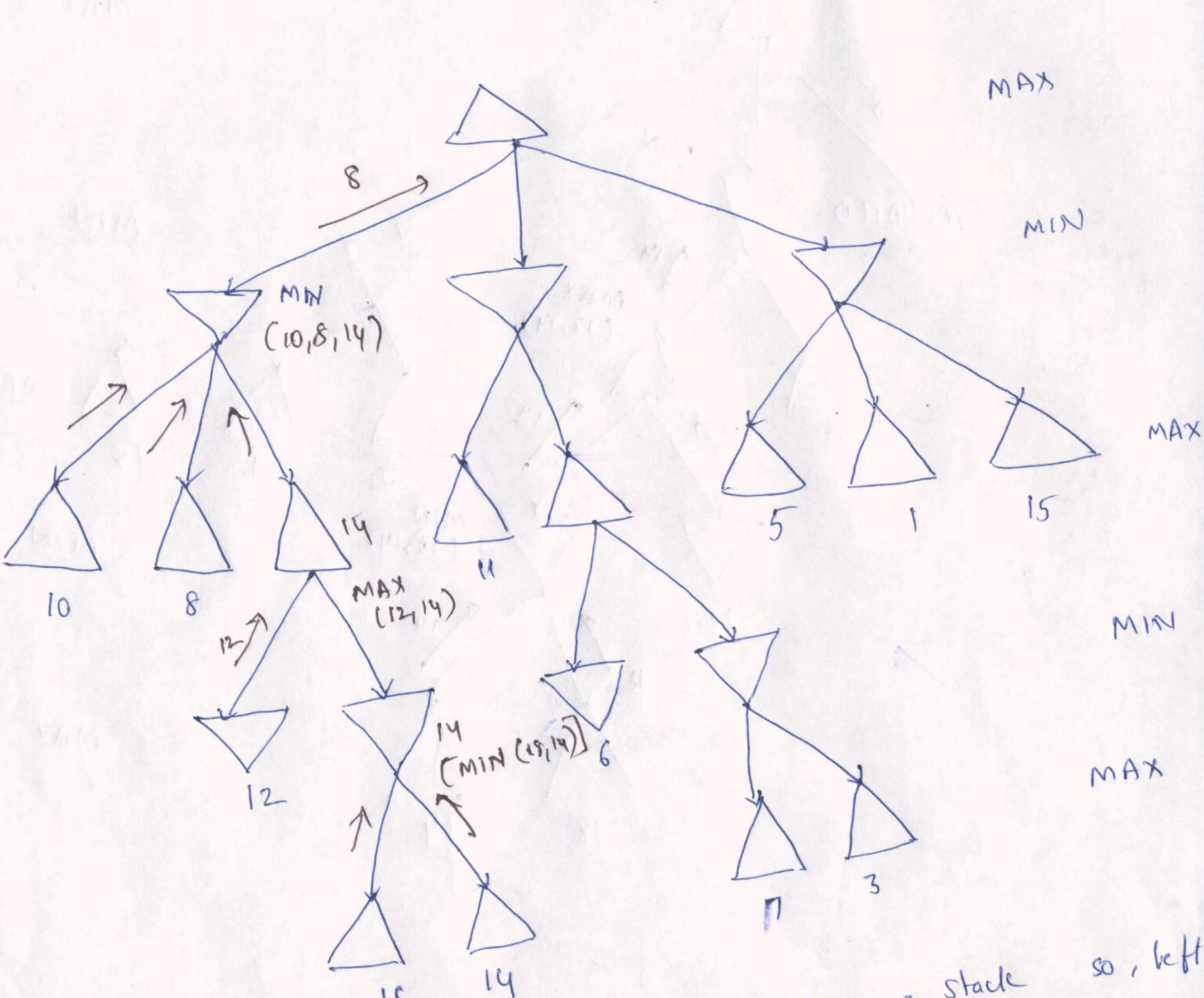
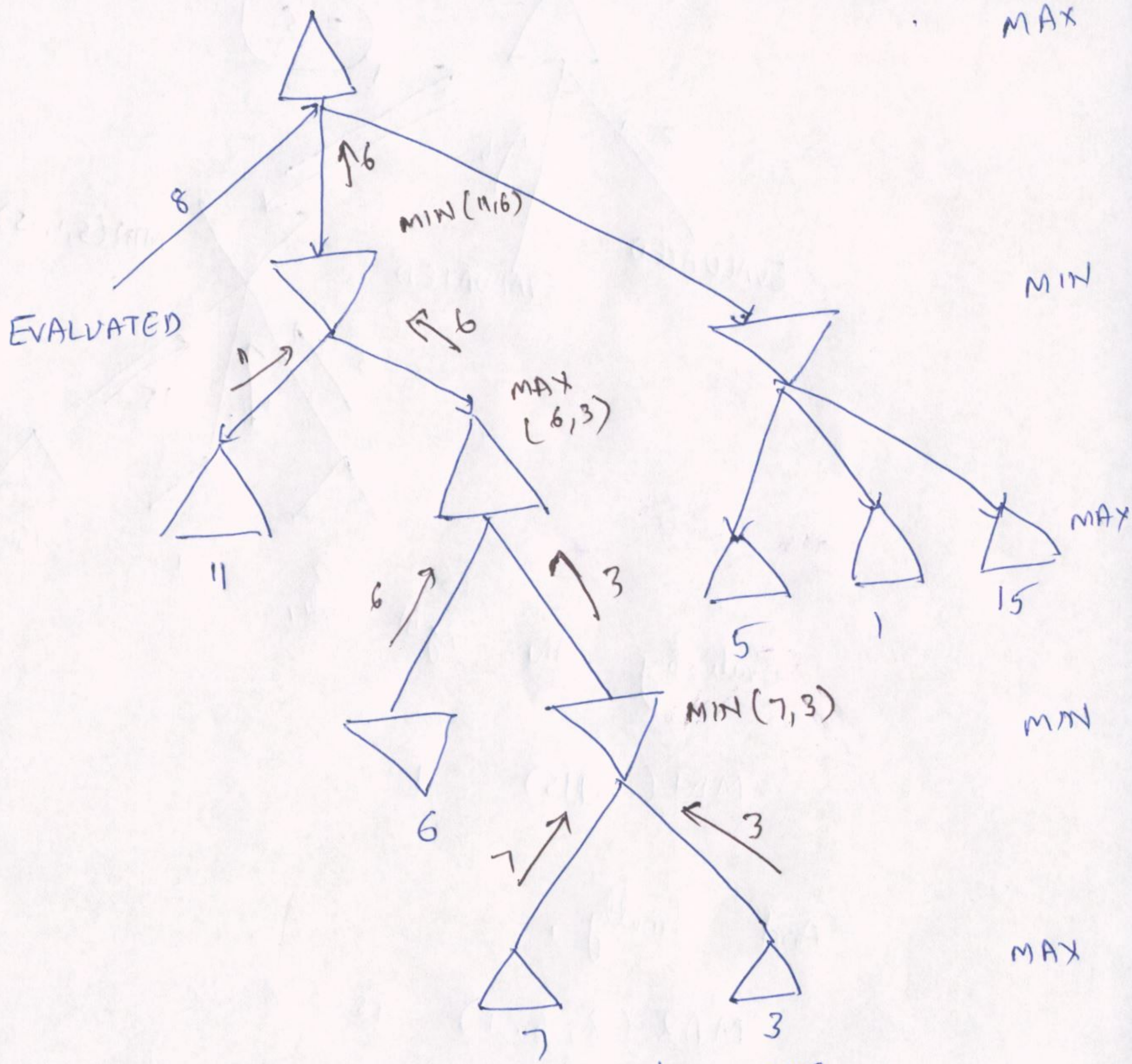
① MIN-MAX ALGORITHM



→ this algorithm uses DFS search. So using a stack so, left most nodes will be visited first

MIN takes 10, 8 and right most node which is still has to be evaluated

on traversing the right most, it is MAX's turn.
so, takes MAX (12, right_node).
on traversing to the MAX right-node,
MIN (18, 14) ⟶ 14
MAX (12, 14) ⟶ 14 ; MIN(10, 8, 14) = 8



Traverse through the middle node now.

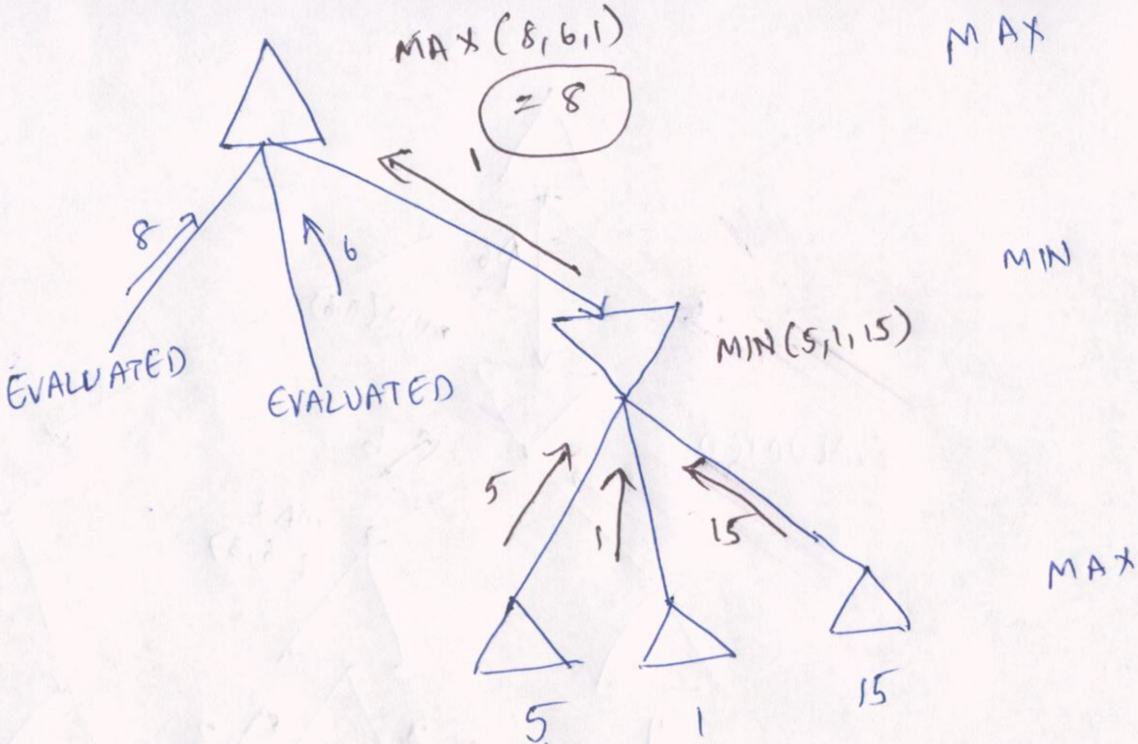
MIN (11, right-node1)

MAX (6, right-node2)

Evaluate right-node2 now i.e MIN(7, 3) = 3

Now MAX (6, 3) = 6

And finally MIN (11, 6) = 6



Evaluating the right node,

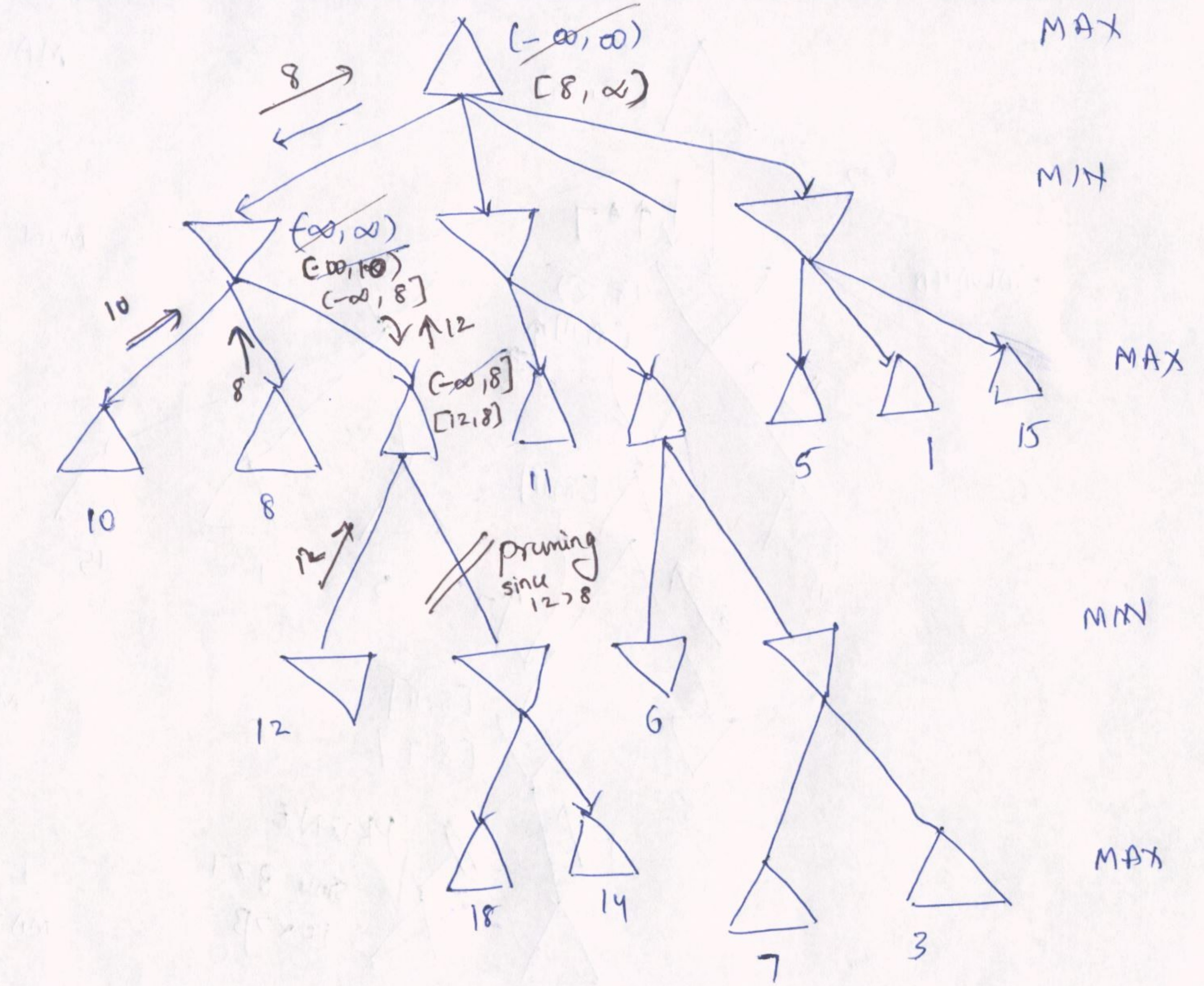
MIN (5,1,15) = 1

And finally,

MAX (8, 6, 1) is 8

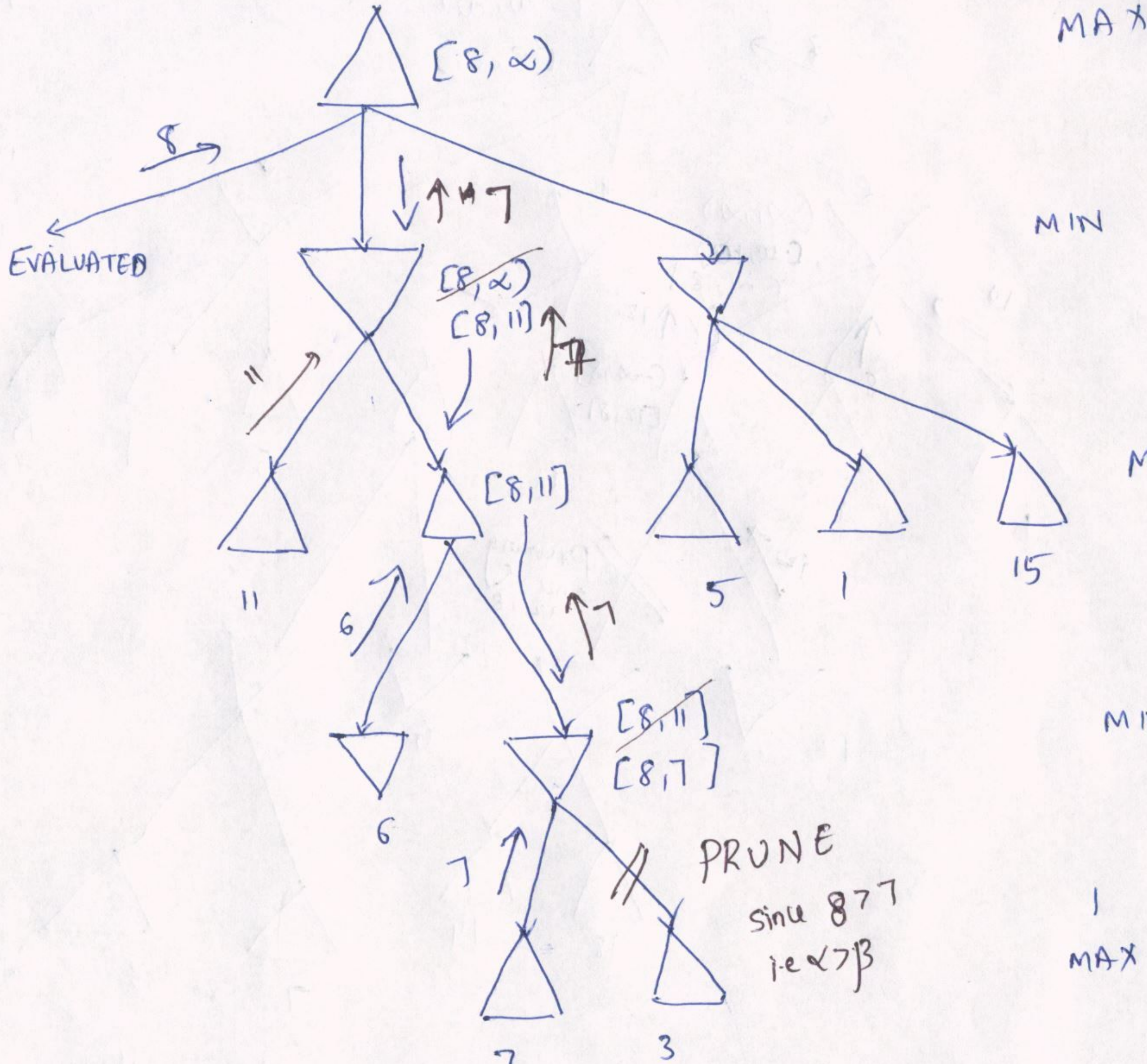
8 is the maximum utility that MAX can achieve.

2



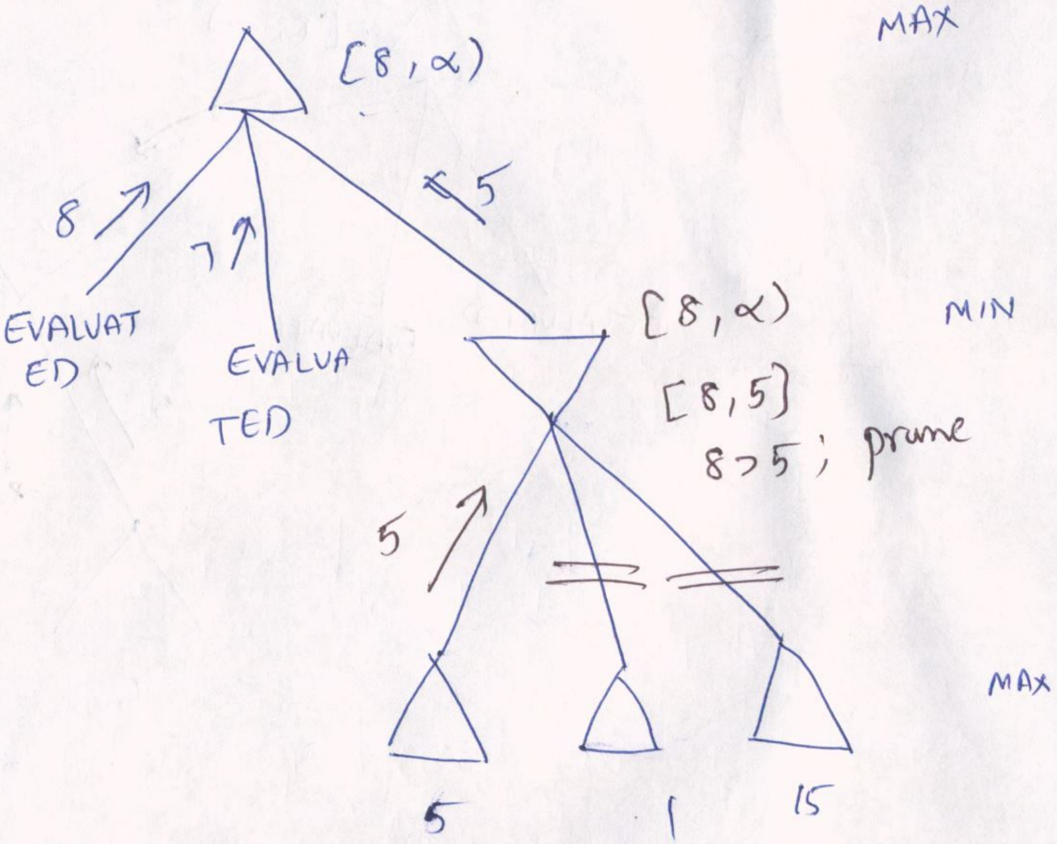
MAX prunes the branch with three nodes, since $V \geq \beta$ i.e $12 \geq 8$ and MIN has ~~8~~ $\alpha = 8$

And MAX sends 12 as the value and MIN has 8 already, so it does not get updated.



Here, we pruned one MAX node with value 3,

since 8 > 7 ; MIN prunes node 3.



Finally, we pruned 4 branches with 6 nodes.

Q(3) A: { 4, 5, 6, 7, 8 }

B: { 10, 20, 30, 40 }

C: { 2, 3, 4 }

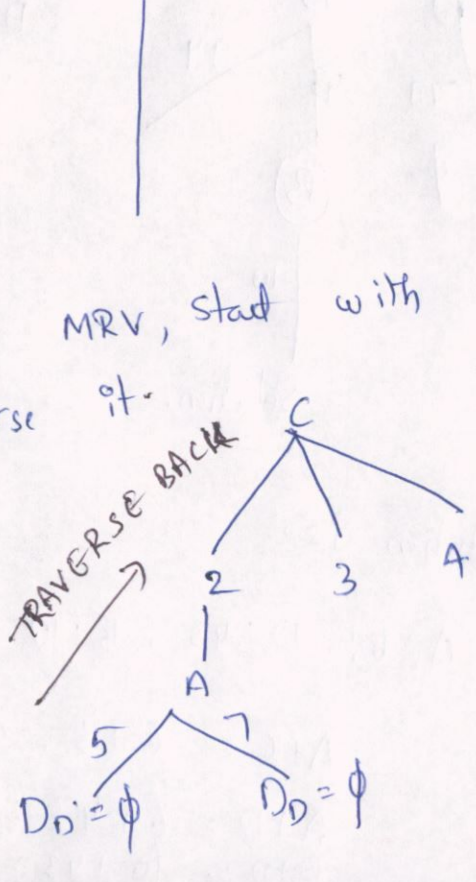
D: { 28, 43, 56, 77, 94, 114 }

Constraints are A + C = odd ; A + D = square of an integer

B + D < 60

Domain	C = 2	A = 5	A = 7
A: {4, 5, 6, 7}	{5, 7}	5	7
B: {10, 20, 30, 40}	{10, 20, 30, 40}	{10, 20, 30, 40}	{10, 20, 30, 40}
C: {2, 3, 4}	2	2	2
D: {28, 43, 56, 77, 94, 114}	{28, 43, 56, 77, 94, 114}	{ ϕ	ϕ

Based on MRV, start with ~~A~~ C ; since A is MRV, traverse it.

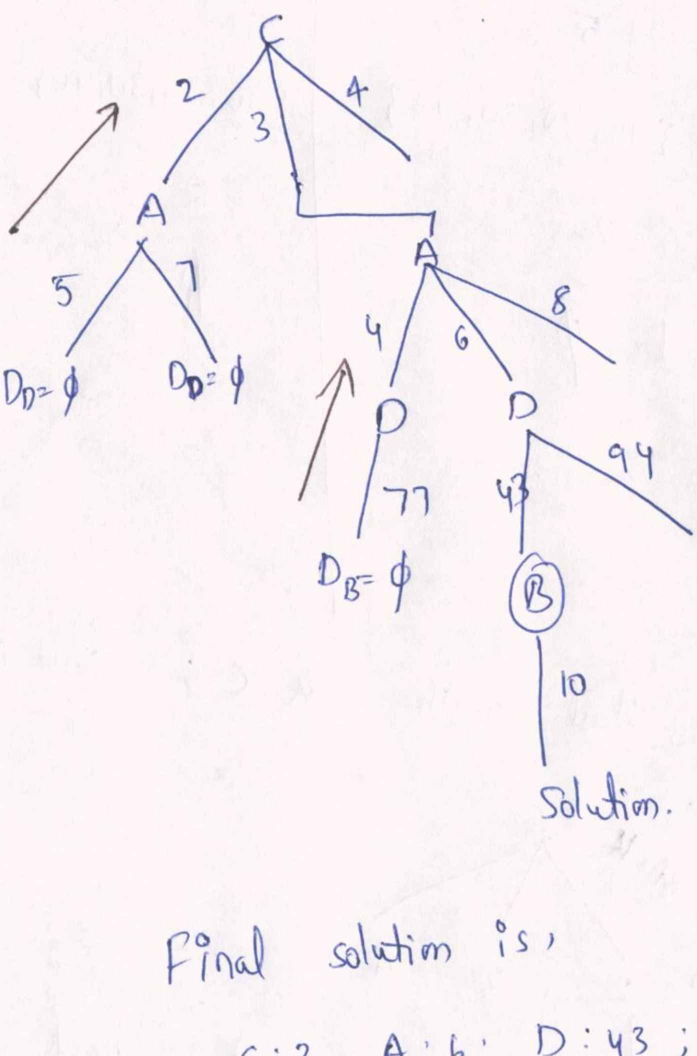


Lets check with

C = 3

	C=3	A=4	D=77	A=6	D=43	B=10
A: {4,5,6,7,8}	{4,6,8}	6	4	6	6	6
B: {10,20,30,40}	{10,20,30,40}	{10,20,30,40}	ϕ	{10,20,30,40}	{10}	10
C: {2,3,4}	3	3	3	3	3	3
D: {28,43,56,77,94,114}	{28,43,56,77,94,114}	{77}	77	{43,94}	43	43

On selecting C = 3; we have A with 4, 6, and 8 values.
with A as 4, D as 77, we are getting an empty set for B.
Traverse back to A = 6, D with 43, 94 are the values which passes all Constraints.



Solution.

Final solution is,

C : 3, A : 6; D : 43 ; B : 10

A + C = 6 + 3 = odd

A + D = 6 + 43 = 49 = 7^2 = square of integer

B + D = 10 + 43 = 53 < 60

(4) TWO + TWO = FOUR

$F = 1;$

$O + O = 10 + R;$ $W + W + 1 = 10 + U;$

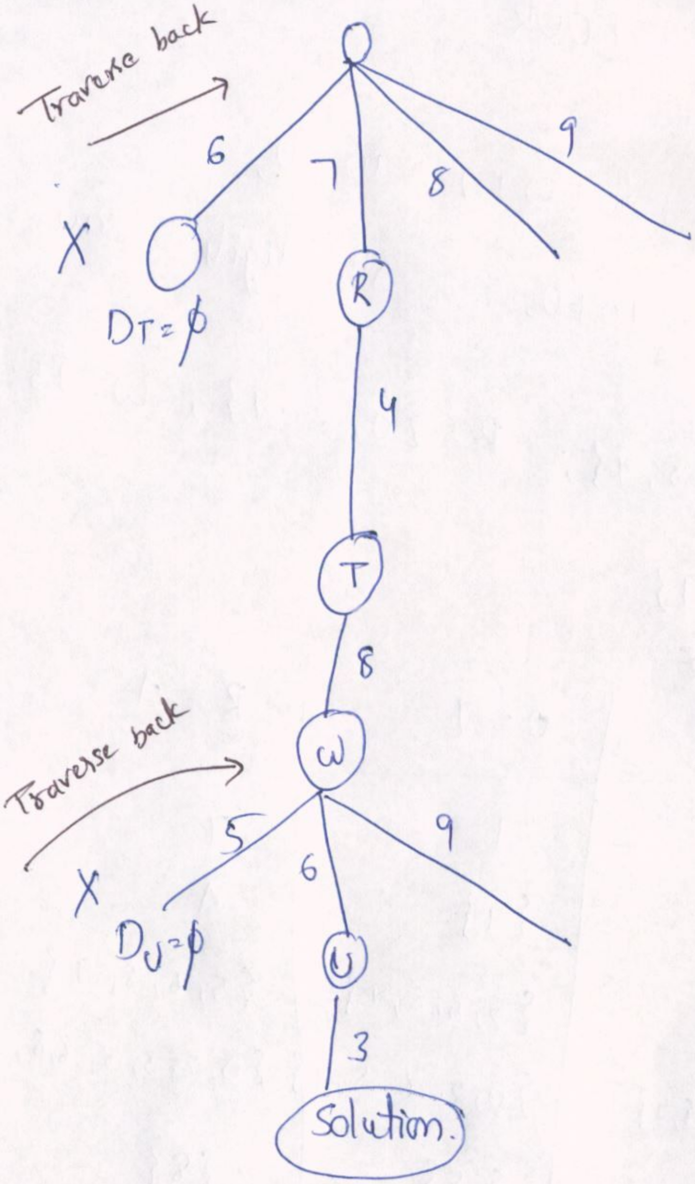
$T + T + 1 = 10 + O;$ All digits are distinct.

$O: \{6, 7, 8, 9\};$ $R: \{0, 2..9\};$ $W: \{5..9\};$ $U: \{0, 2..9\}$

$T: \{5..9\}$

Domain	O=6 ← Traverse back	O=7	R=4	T=8
O: {6,7,8,9}	6	7	7	7
R: {0,2..9}	{2}	{4}	4	4
W: {5..9}	{5,7,8,9}	{5,6,8,9}	{5,6,8,9}	{5,6,9}
U: {0,2..9}	{0,2,3,4,5,7,8,9}	{0,2..6,8,9}	{0,2,3,5,6,8,9}	{0,2,3,5,6,9}
T: {5..9}	ϕ	{8}	{8}	8

	W=5 (Traversing Back)	W=6	U=3 ← Solution.
O	7	7	7
R	4	4	4
W	5	6	6
U	ϕ	{3}	3
T	8	8	8



T=8; W=6; O=7; U=3; F=1; R=4

TWO + TWO = FOUR

$$\begin{array}{r} 867 \\ 867 \\ \hline 1734 \end{array}$$

F O U R