23K-0006

QUESTION 2:

▸ Dry-Run:-

★ generation(0) → initial population:

{ [1, 2, 1, 2, 3, 1, 2], [3, 1, 2, 2, 2, 3, 3],
[2, 2, 2, 1, 1, 2, 2], [1, 3, 1, 3, 2, 1, 1],
[2, 1, 3, 2, 3, 3, 2], [3, 2, 3, 1, 1, 1, 3] }

★ sample calculation of fitness:-

• Individual fitness 1: [1, 2, 1, 2, 3, 1, 2]

→ T1 (5 hrs) → F1 → Cost: 5 x 10 = 50
→ T2 (8 hrs) → F2 → 8 x 14 = 112
→ T3 (4 hrs) → F1 → 4 x 8 = 32
→ T4 (7 hrs) → F2 → 7 x 10 = 70
→ T5 (6 hrs) → F3 → 6 x 12 = 72
→ T6 (3 hrs) → F1 → 3 x 9 = 27
→ T7 (9 hrs) → F2 → 9 x 12 = 108

used time:

F1: 5 + 4 + 3 = 12 hrs
F2: 8 + 7 + 9 = 24 hrs
F3: 6 hrs

total = 50 + 112 + 32 + 70 + 72 + 27 + 108 = ~~447~~ 471

penalty = 0

fitness = 1 / (471 + 0) = 0.002122

∴ same for rest - 6

Ind	Cost	Penalty	Fitness
1	471	0	0.002122
2	511	0	0.001957
3	515	0	0.001942
4	499	0	0.002004
5	514	0	0.001946
6	495	0	0.002020

CamScanner

* Roulette Wheel:

→ computing probabilities

total fitness = ~0.012

Ind 1 → 0.002122/0.012 = 17.7 %

Ind 2 → 16.3

Ind 3 → 16.2

Ind 4 → 16.7

Ind 5 → 16.2

Ind 6 → 16.8

* Cross Over:

- selecting a pair for cross over (randomly):
  - Ind 4 & Ind 2

Ind 4 → [1, 3, 1, 3, 2, 1, 1] , Ind 6 → [3, 1, 2, 2, 2, 3, 3]

- cross over at point 4:

  C1 : [1, 3, 1, 3, 2, 3, 3]

  C2 : [3, 1, 2, 2, 2, 1, 1]

* Mutation:

swapping pos 1 & 5

Mutated Ch1 : [3, 3, 1, 2, 2, 3, 3]

Mutated Ch2 : [1, 1, 2, 3, 2, 1, 1]

Keep generating for rest of the generation :)

QUESTION 4:

O's turn

X		
X		
O	O	

X's turn

X		
X		
O	O	X

V = 110

X		
X	X	
O	O	

V = 110

X	X	
X		
O	O	

V = 10

X		X
X		
O	O	

V = 20

X		
X		X
O	O	

V = 10

X		
X	O	
O	O	X

V = -180

X		
X		O
O	O	X

V = 90

X	O	
X		
O	O	X

V = 10

X		O
X		
O	O	X

V = 0

X	X	
X	O	
O	O	

V = -100

X	X	
X		O
O	O	

V = 10

X	X	O
X		
O	O	

V = -190

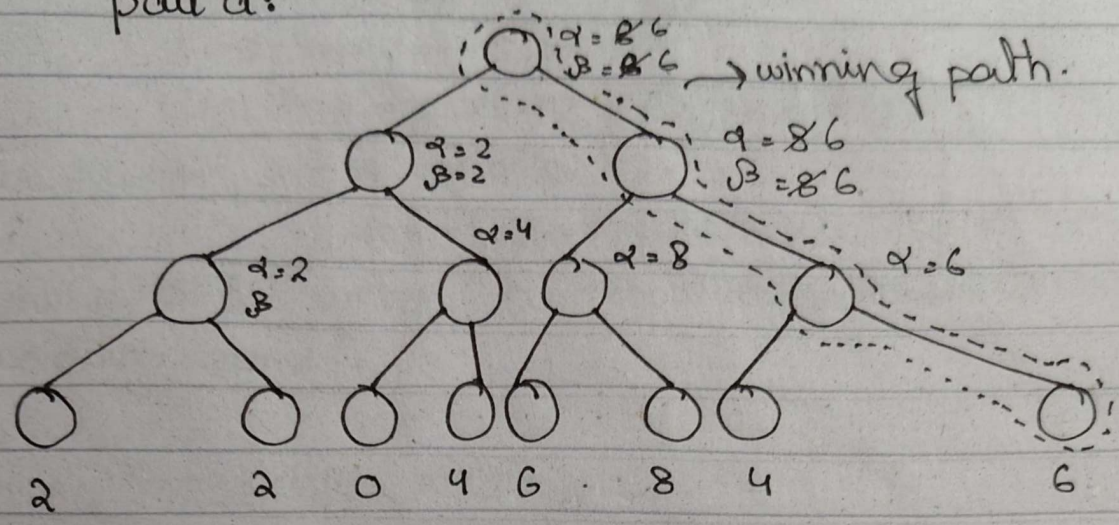
X	X	
X		
O	O	O

V = -1000

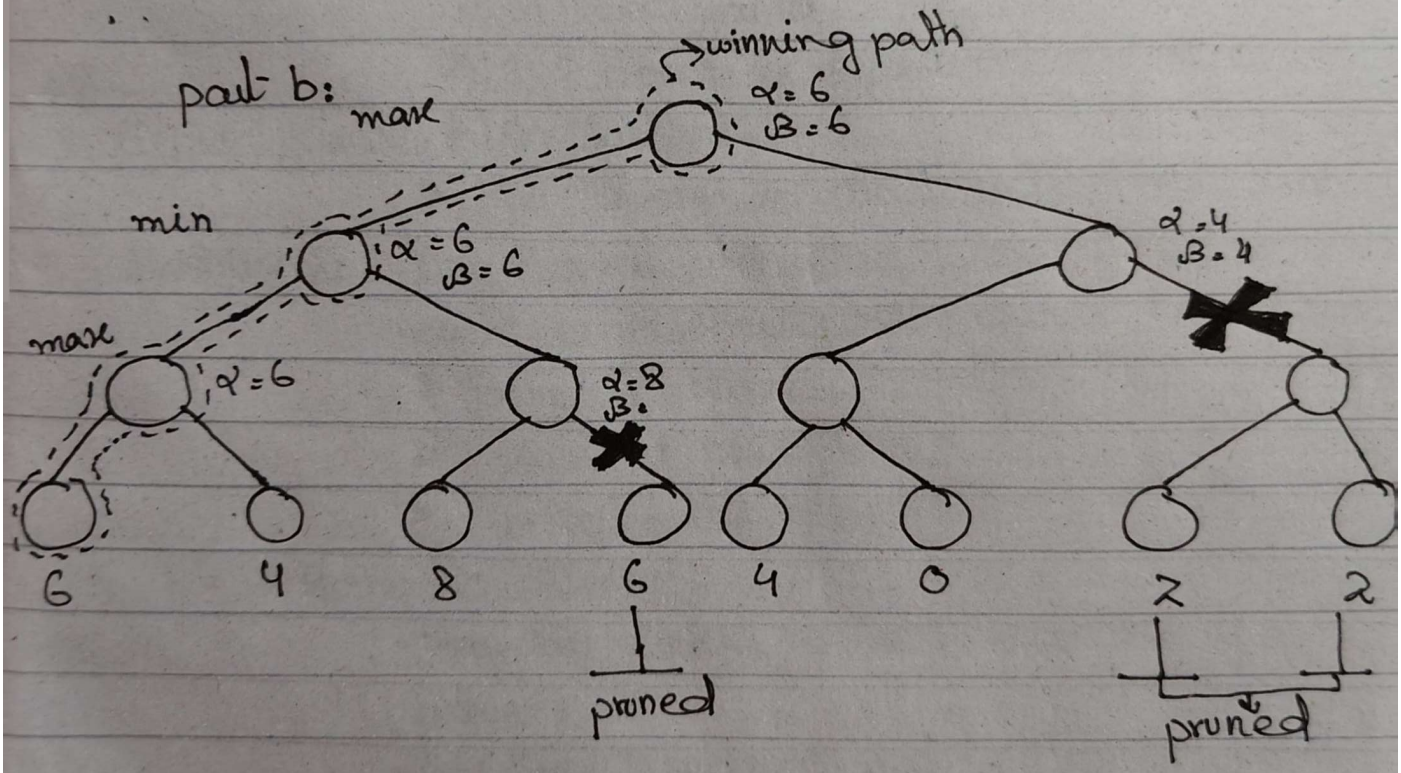
X will choose the left most block to maximize

Q5)

part a:



part b:



Question 06:-

Part-(a): Game Model:-

→ players & objective:

- max(defender/AI system): tries to min the success & impact of attacks.
- min (attacker/hacker): tries to maximize damage to the system by exploiting vulnerabilities & evading defenses.

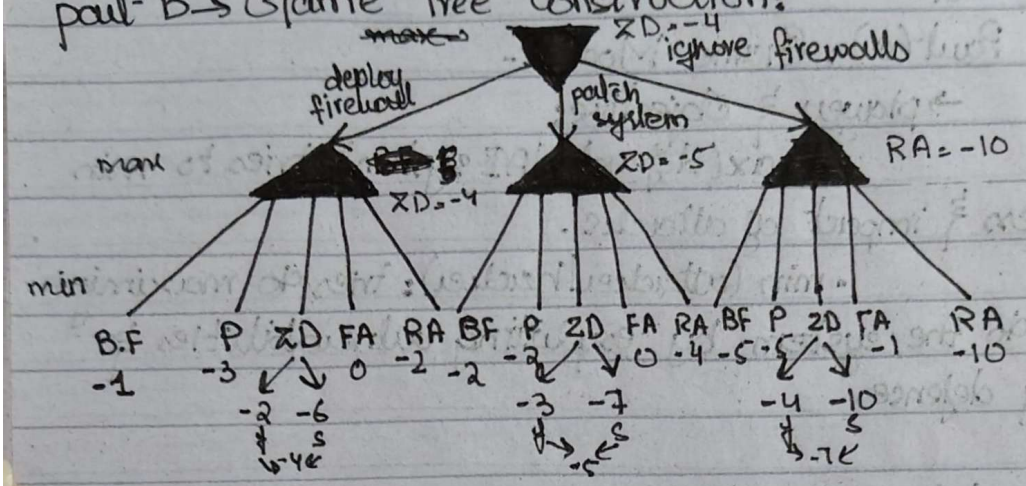
→ decision making:

- turn based game:
  - Max chooses a defense action first (Deploy firewall, Patch System)...
  - Min chooses an attack action (Brute force, Phishing, Zero-Day Exploits, fake attack, real attacks).
- each player strategically choose based on:
  - Expected opponent's action
  - potential payoffs
  - probabilistic outcomes

→ Stochastic Elements:

- zero-day exploit has a 50% chance of success.
- It introduces chance nodes into the game tree.
- this forces the defender to account for expected utility, not just worst case damage

part b → Game Tree Construction:

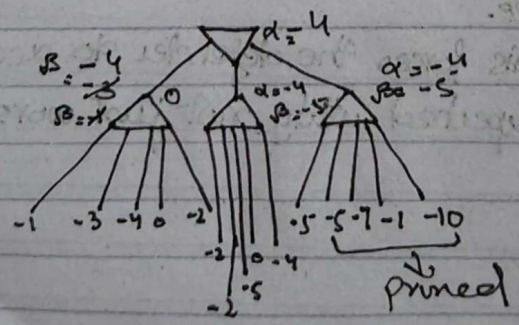


- utility values to represent the defender's losses.
  - → lower utility = worst outcome for defender
  - → higher utility = less harmful or (better) outcome.
  - → rep on (0 → -10) ...
    - 0 → neutral, -10 → worst
  - → zeroday → 50% failure → -2
    - 50% success → -6

part c → Min Max & alpha beta pruning:

1) ~~M1 → worst~~.
   - deploy firewall → worst ⇒ -4
   - patch system → worst ⇒ -5
   - ignore → worst ⇒ -10
- Best move for Max: M1 (deploy firewall) with utility: -4 (dry run it above).

2)



- the values pruned are β = -5 < α = -4
- α = -4 (Max best so far)
- β = (min's best) varies
- pruned: all after brute force in firewalls ignore.

part d→ Expectimax and Probabilistic Attacks:

1) Expected value of X-day exploit:-

Success Rate: 50%

Under Max's options:

firewall	$0.5(-6)+0.5(-2)=-4$
patch system	$0.5(-7)+0.5(-3)=-5$
Ignore Alerts	$0.5(-10)+0.5(-4)=-7$

2) • under expectimax, max chooses the action with highest expected utility, rather than minimizing worst case damage.

• max should still choose "Deploy Firewall", as it results in the lowest expected lost.

CS CamScanner