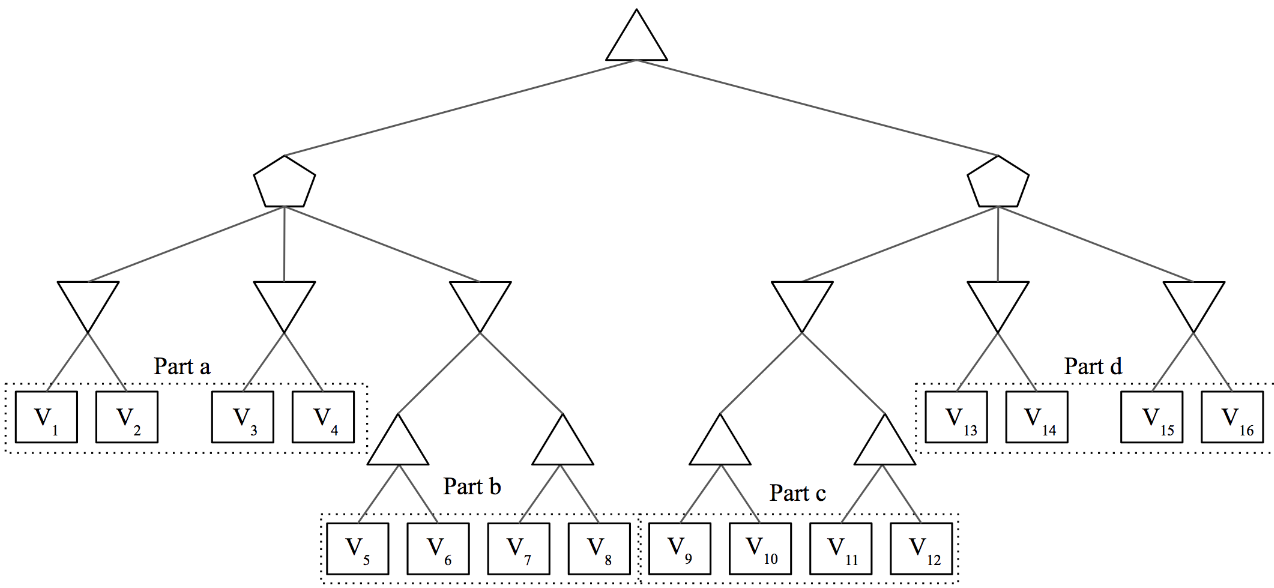
Introduction to AI - Homework 6 Solved by Dino Meng

Pages 1-2: responses
Pages 3-6: detailed resolution

Q1. MedianMiniMax

You're living in utopia! Despite living in utopia, you still believe that you need to maximize your utility in life, other people want to minimize your utility, and the world is a 0 sum game. But because you live in utopia, a benevolent social planner occasionally steps in and chooses an option that is a compromise. Essentially, the social planner (represented as the pentagon) is a median node that chooses the successor with median utility. Your struggle with your fellow citizens can be modelled as follows:



There are some nodes that we are sometimes able to prune. In each part, mark all of the terminal nodes such that **there exists a possible situation** for which the node **can be pruned**. In other words, you must consider **all** possible pruning situations. Assume that evaluation order is **left to right** and all V_i's are **distinct.**

Note that as long as there exists ANY pruning situation (does not have to be the same situation for every node), you should mark the node as prunable. Also, alpha-beta pruning does not apply here, simply prune a sub-tree when you can reason that its value will not affect your final utility.

(a)
- [ ] V_1
- [ ] V_2
- [ ] V_3
- [ ] V_4
- [x] None

(b)
- [ ] V_5
- [x] V_6
- [x] V_7
- [x] V_8
- [ ] None

(c)
- [ ] V_9
- [ ] V_{10}
- [x] V_{11}
- [x] V_{12}
- [ ] None

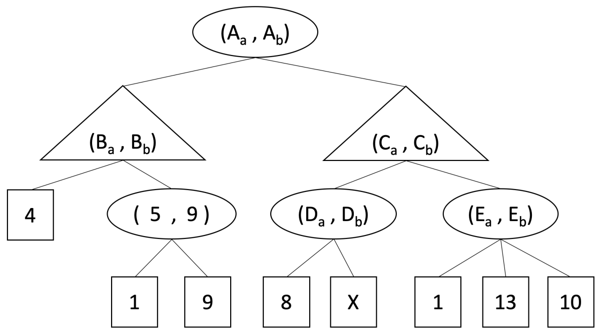
(d)
- [ ] V_{13}
- [x] V_{14}
- [x] V_{15}
- [x] V_{16}
- [ ] None

Q2. Games

Alice is playing a two-player game with Bob, in which they move alternately. Alice is a maximizer. Although Bob is also a maximizer, Alice believes Bob is a minimizer with probability 0.5, and a maximizer with probability 0.5. Bob is aware of Alice's assumption.

In the game tree below, square nodes are the outcomes, triangular nodes are Alice's moves, and round nodes are Bob's moves. Each node for Alice/Bob contains a tuple, the left value being Alice's expectation of the outcome, and the right value being Bob's expectation of the outcome.

Tie-breaking: choose the left branch.



(a) In the blanks below, fill in the tuple values for tuples (B_a, B_b) and (E_a, E_b) from the above game tree.

$(B_a, B_b) = ($ 5 , 9 $)$

$(E_a, E_b) = ($ 7 , 13 $)$

(b) In this part, we will determine the values for tuple (D_a, D_b).

(i) $D_a =$ ○ 8 ○ X ○ 8+X ⊗ 4+0.5X ○ min(8,X) ○ max(8,X)

(ii) $D_b =$ ○ 8 ○ X ○ 8+X ○ 4+0.5X ○ min(8,X) ⊗ max(8,X)

(c) Fill in the values for tuple (C_a, C_b) below. For the bounds of X, you may write scalars, ∞ or $-\infty$.
If your answer contains a fraction, please write down the corresponding **simplified decimal value** in its place. (i.e., 4 instead of $\frac{8}{2}$, and 0.5 instead of $\frac{1}{2}$).

1. If $-\infty <$ X $<$ 6 , $(C_a, C_b) = ($ 7 , 13 $)$
2. Else, $(C_a, C_b) = ($ 4+0.5(x) , max(x , 8 $))$

(d) Fill in the values for tuple (A_a, A_b) below. For the bounds of X, you may write scalars, ∞ or $-\infty$.
If your answer contains a fraction, please write down the corresponding **simplified decimal value** in its place. (i.e., 4 instead of $\frac{8}{2}$, and 0.5 instead of $\frac{1}{2}$).

1. If $-\infty <$ X $<$ 6 , $(A_a, A_b) = ($ 6 , 13 $)$
2. Else, $(A_a, A_b) = ($ 4.5+0.25(x) , max(x , 9 $))$

(e) When Alice computes the left values in the tree, some branches can be pruned and do not need to be explored. In the game tree graph above, put an 'X' on these branches. If no branches can be pruned, write "Not Possible" below. Assume that the children of a node are visited in left-to-right order and that you should not prune on equality.

Not possible, with "chance" nodes
it's not possible to use pruning
techniques

Introduction to Artificial Intelligence

Homework 6 Resolution by Dino Meng [SM3201466]

Q1. MedianMiniMax

Part A. We do not prune any of the nodes of this part, as they are all necessary.

Part B. We have two pruning situations: one where we can prune V_6 and V_8, and another where we can prune V_7, V_8. Assuming the other two parallel minimizer' values are known (as we already explored them), and assume without loss of generality that the second minimizer is the "median minimizer". 1. If the visiting minimizer (the one above group B) would be the "high" minimizer, then it would be sufficient to visit the first values of each maximizer and find out their values are "sufficiently high", and the values cannot get lower. 2. If the visiting minimizer were to be the "low minimizer", then it is sufficient to visit the first maximizer and find out that its value is "sufficiently low" (by exploring both of the left-side maximizer's nodes) and since that the values cannot get higher, ignore the nodes associated to the other maximizer.

The figure below represents a possible configuration of the described situations:

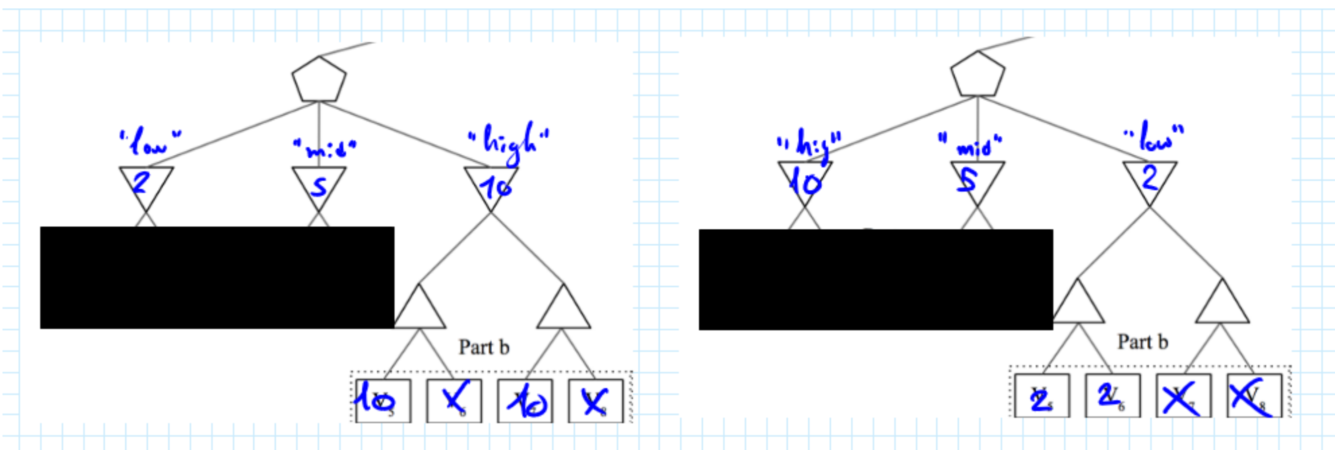


Figure 1: PART B, Two Cases

PART C. Assume that the left "median" has value 10, thus the root node has value at least ≥ 10. Suppose that by exploring V_9, V_{10} we obtain an upper bound on the left minimizer (for example, set $V_9 = V_{10} = 5$). Then we immediately notice that V_{11}, V_{12} have no effect on the outcome of root's value, in fact:

- If the other two minimizers have value less than 10, then the "medianizer" will always have a value less than 10, confirming the lower bound of the root node
- If one minimizer has a value less than 10 and another a value greater than 10, then either the first minimizer (with value ≤ 5) will be choosen as the median or the other minimizer with value less than 10, which causes no change anyways
- If the other two minimizers have a value greater than 10, then the minimum minimizer with value greater than 10 will be choosen and change the root node's lower bound; however, the value cannot come from the first minimizer, meaning that V_{11}, V_{12} had no effect on this outcome at all.

Therefore we can prune V_{11}, V_{12}.

PART D. Assume, like before, that the left median has value 10 meaning that root node has value ≥ 10. Suppose that the left-side minimizer has value 1. As we explore V_{13} and V_{14} (for example let's set them to 2), we immediately notice that we can discard V_{15}, V_{16} because:

- If the right-side minimizer had value less than 1, then the minimizer with value 2 would be chosen, which does not affect the root node

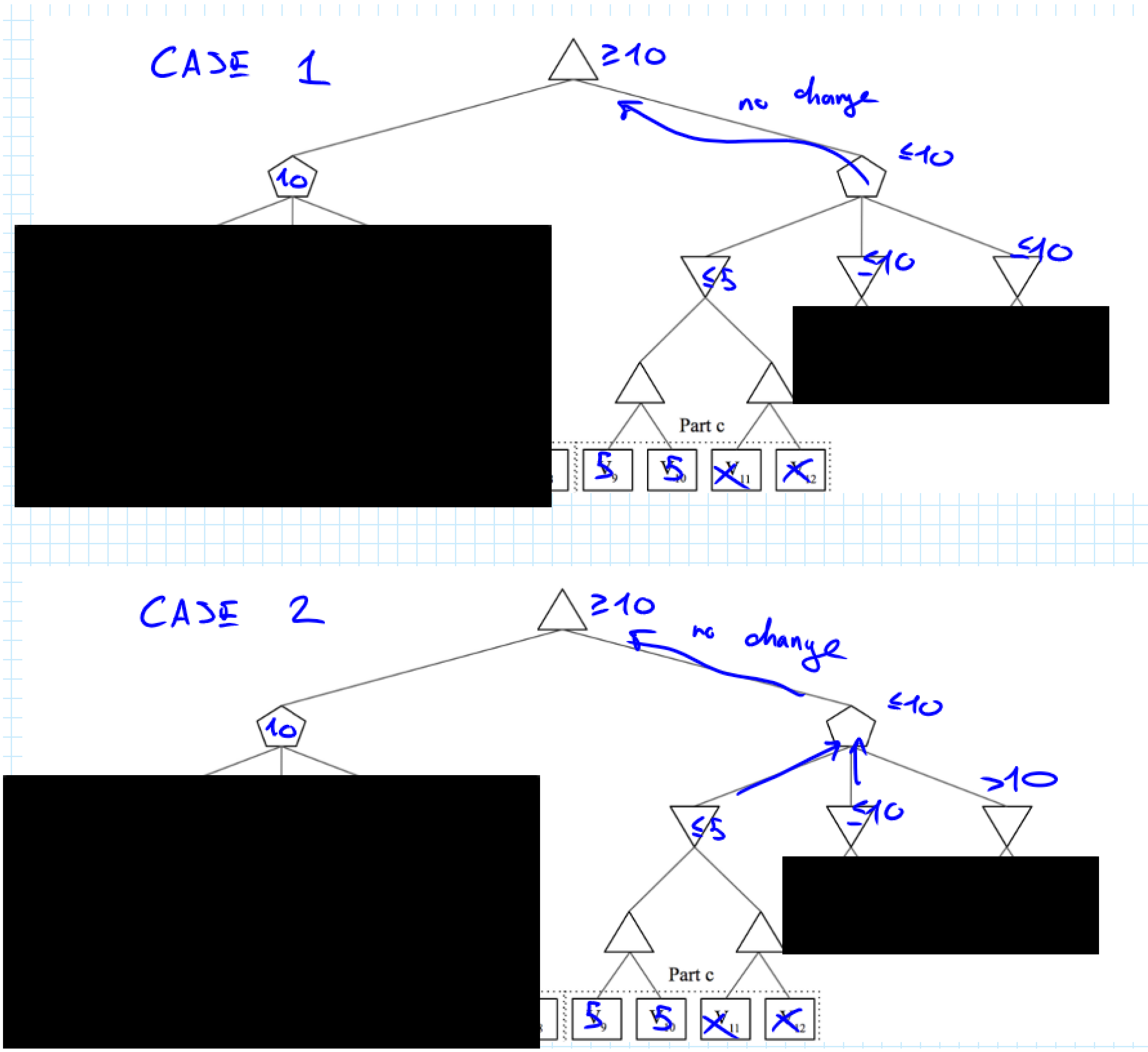


Figure 2: PART C, Case 1, 2

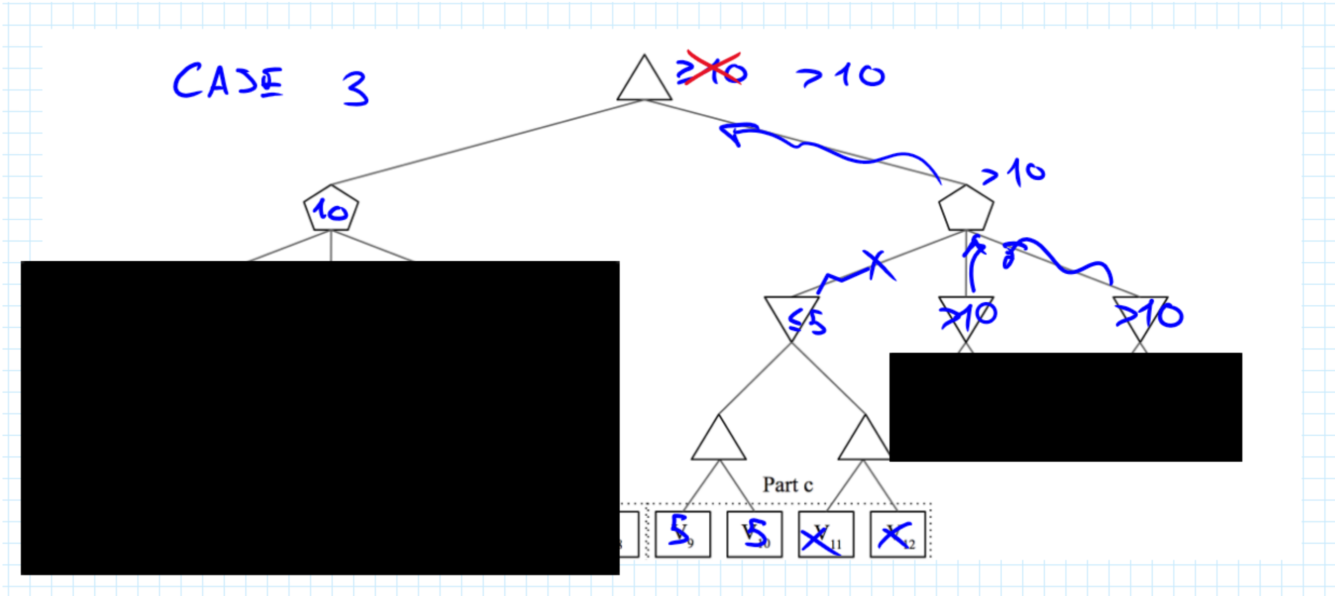


Figure 3: PART C, Case 3

- If the right-side minimizer had value greater than 2, then the minimizer with value 1 would be chosen, which does not affect the root node as well
- If the right-side minimizer had a value between 1 and 2, this minimizer would have been chosen which does not affect the root since it's value is lower than 10

Therefore, as the right-side minimizer's value is uninfluential, we can discard their associated nodes. Moreover notice how also V_{14} can be discarded since that if it were an even lower value (for instance, -1) then the same situation described above would have happened, except that the left-side and center minimizers "swap roles" with each other.

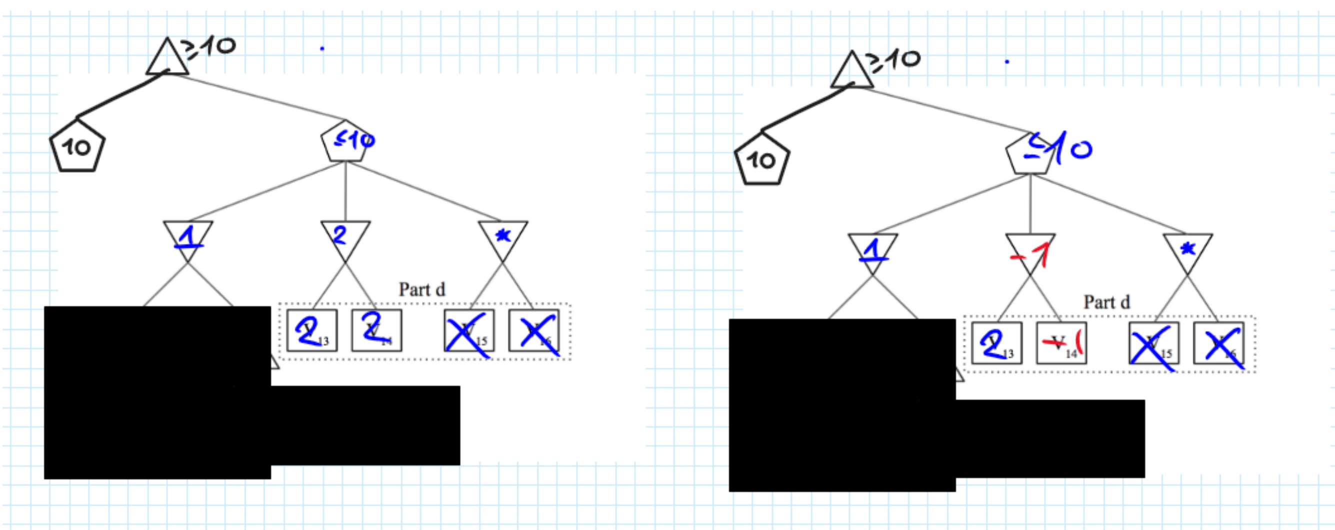


Figure 4: Part D

Q2. Games

Calculated values:

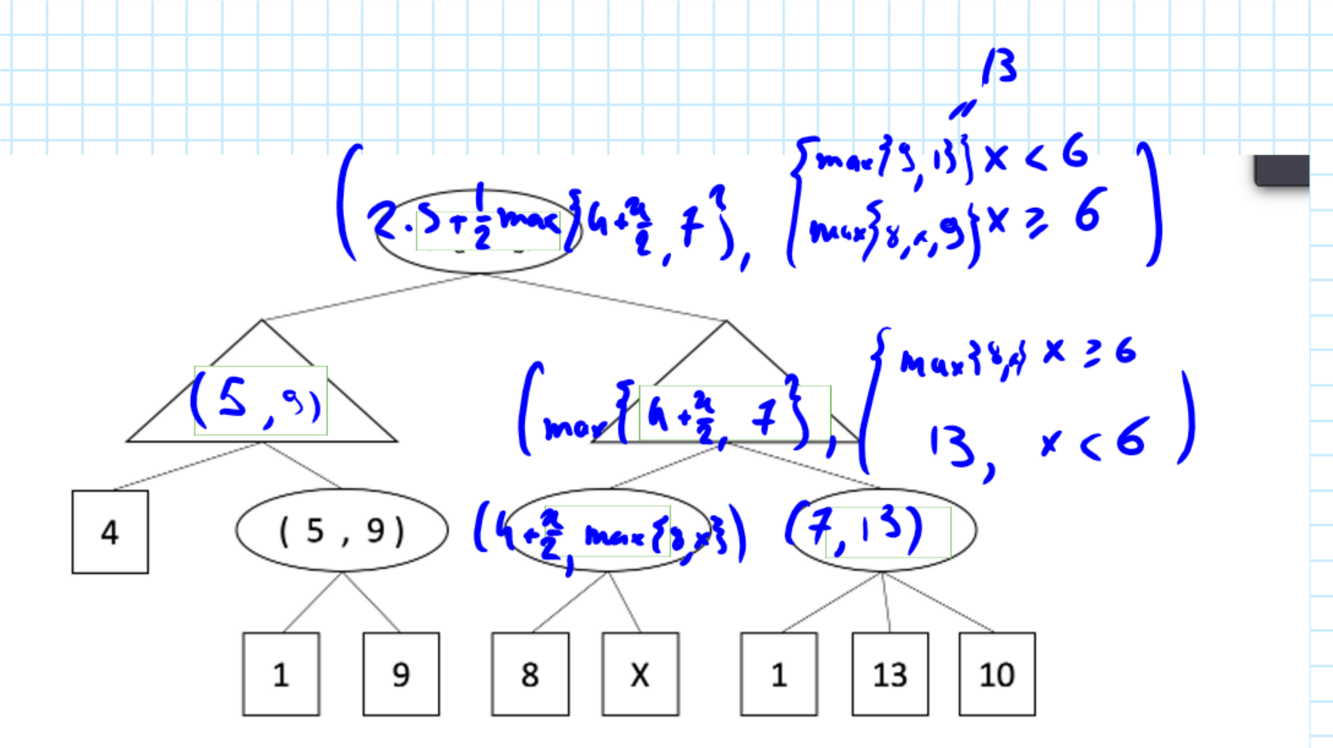


Figure 5: Calculated Adversarial Search Tree

To answer the questions, just plug in the values of the figure above. Note that C_b depends on the value X as Bob knows how Alice estimates her value, so his value depends on “decision” made by Alice.

Pruning is not possible for Alice’s “branch” as there are “chance nodes” and we have made no assumptions about the range of values prior.