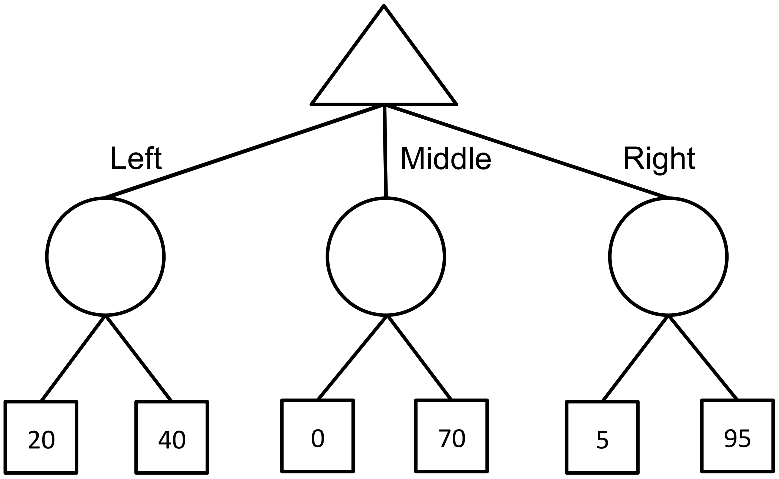
CS 188
Spring 2019

Midterm Review Games Solutions

Q1. Bounded Expectimax

(a) **Expectimax.** Consider the game tree below, where the terminal values are the *payoffs* of the game. Fill in the expectimax values, assuming that player 1 is maximizing expected payoff and player 2 plays uniformly at random (i.e., each action available has equal probability).



(b) Again, assume that Player 1 follows an expectimax strategy (i.e., maximizes expected payoff) and Player 2 plays uniformly at random (i.e., each action available has equal probability).

(i) What is Player 1's expected payoff if she takes the expectimax optimal action?

50

(ii) Multiple outcomes are possible from Player 1's expectimax play. What is the worst possible payoff she could see from that action?

5

(c) Even if the average outcome is good, Player 1 doesn't like that very bad outcomes are possible. Therefore, rather than purely maximizing expected payoff using expectimax, Player 1 chooses to perform a modified search. In particular, she only considers actions whose worst-case outcome is 10 or better.

(i) Which action does Player 1 choose for this tree?

Left

(ii) What is the expected payoff for that action?

30

(iii) What is the worst payoff possible for that action?

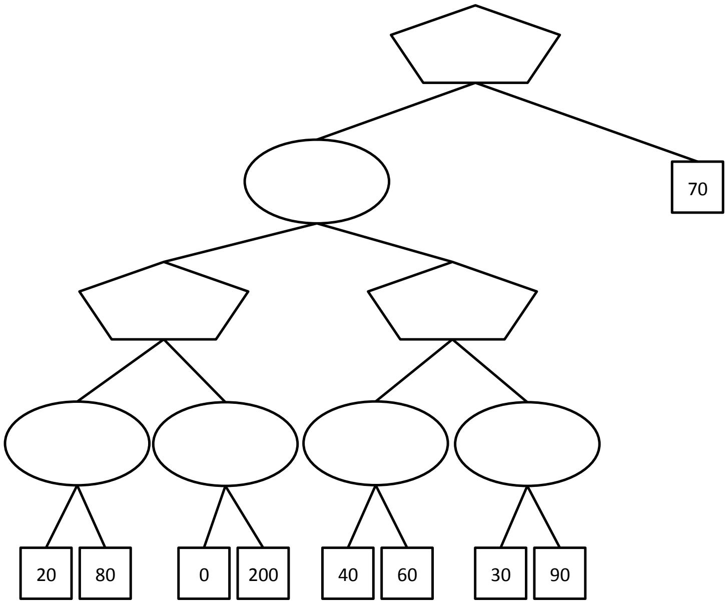
20

(d) Now let's consider a more general case. Player 1 has the following preferences:

- Player 1 prefers any lottery with worst-case outcome of 10 or higher over any lottery with worst-case outcome lower than 10.
- Among two lotteries with worst-case outcome of 10 or higher, Player 1 chooses the one with the highest expected payoff.
- Among two lotteries with worst-case outcome lower than 10, Player 1 chooses the one with the highest worst-case outcome (breaking ties by highest expected payoff).

Player 2 still always plays uniformly at random.

To compute the appropriate values of tree nodes, Player 1 must consider both expectations and worst-case values at each node. For each node in the game tree below, fill in a pair of numbers (e, w). Here e is the expected value under Player 1's preferences and w is the value of the worst-case outcome under those preferences, assuming that Player 1 and Player 2 play according to the criteria described above.



Last expect-layer, $(50, 20)$, $(100, 0)$, $(50, 40)$, $(60, 30)$
Funny max layer on top of lowest expect layer, $(50, 20)$, $(60, 30)$
Expect layer, $(55, 20)$
Funny max at top, $(70, 70)$

(e) Now let's consider the general case, where the lower bound used by Player 1 is a number L not necessarily equal to 10, and not referring to the particular tree above. Player 2 still plays uniformly at random.

(i) Suppose a Player 1 node has two children: the first child passes up values (e_1, w_1), and the second child passes up values (e_2, w_2). What values (e, w) will be passed up by a Player 1 node if

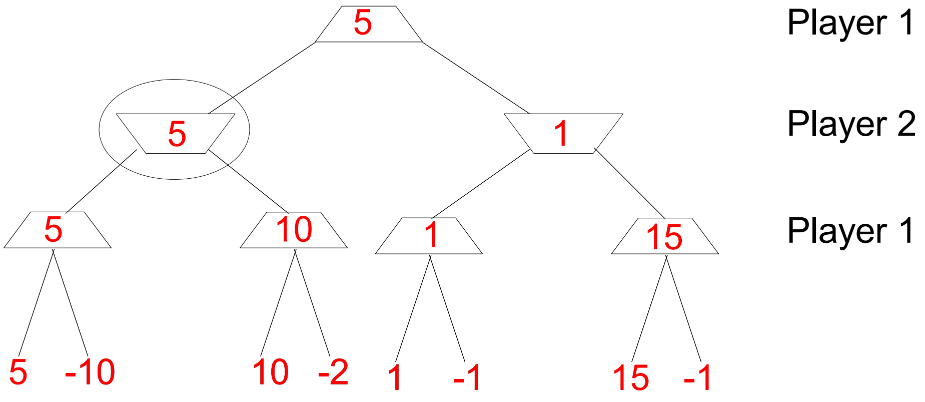
1. $w_1 < w_2 < L$ (e_2, w_2)
2. $w_1 < L < w_2$ (e_2, w_2)
3. $L < w_1 < w_2$ $(max(e_1, e_2), w_{argmax(e_1,e_2)})$

(ii) Now consider a Player 2 node with two children: the first child passes up values (e_1, w_1) and the second child passes up values (e_2, w_2). What values (e, w) will be passed up by a Player 2 node if

1. $w_1 < w_2 < L$ $(mean(e_1, e_2), min(w1, w2))$
2. $w_1 < L < w_2$ $(mean(e_1, e_2), min(w_1, w_2))$
3. $L < w_1 < w_2$ $(mean(e_1, e_2), min(w_1, w_2))$

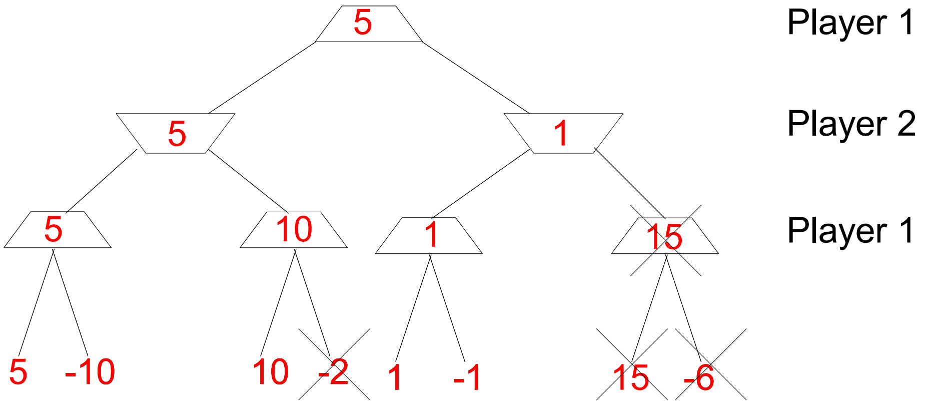
Q2. Minimax and Expectimax

(a) Consider the following zero-sum game with 2 players. At each leaf we have labeled the payoffs Player 1 receives. It is Player 1's turn to move. Assume both players play optimally at every time step (i.e. Player 1 seeks to maximize the payoff, while Player 2 seeks to minimize the payoff). Circle Player 1's optimal next move on the graph, and state the minimax value of the game. Show your work.

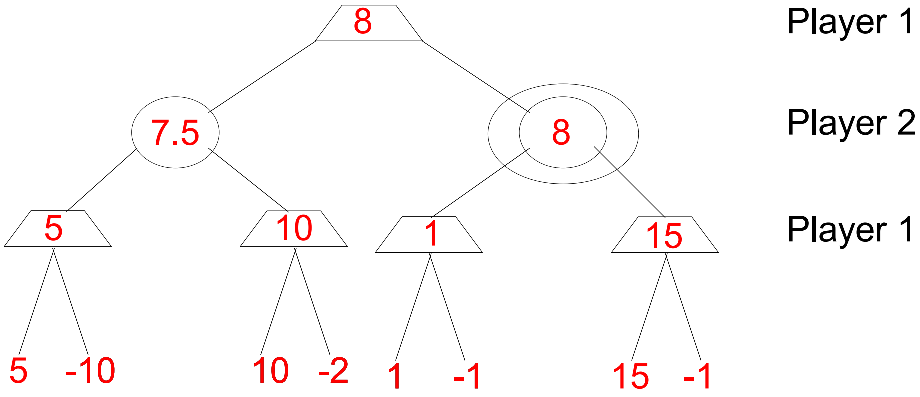


Player 1 should play Left for a payoff of 5.

(b) Consider the following game tree. Player 1 moves first, and attempts to maximize the payoff. Player 2 moves second, and attempts to minimize the payoff. Expand nodes left to right. Cross out nodes pruned by alpha-beta pruning.

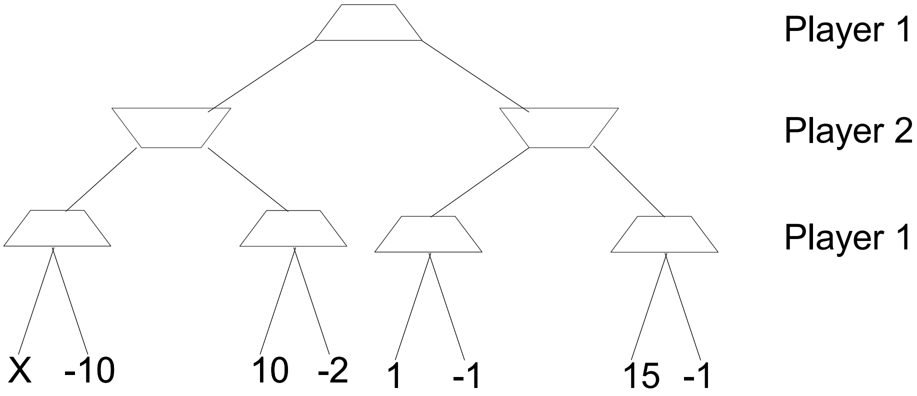


(c) Now assume that Player 2 chooses an action uniformly at random every turn (and Player 1 knows this). Player 1 still seeks to maximize her payoff. Circle Player 1's optimal next move, and give her expected payoff. Show your work.

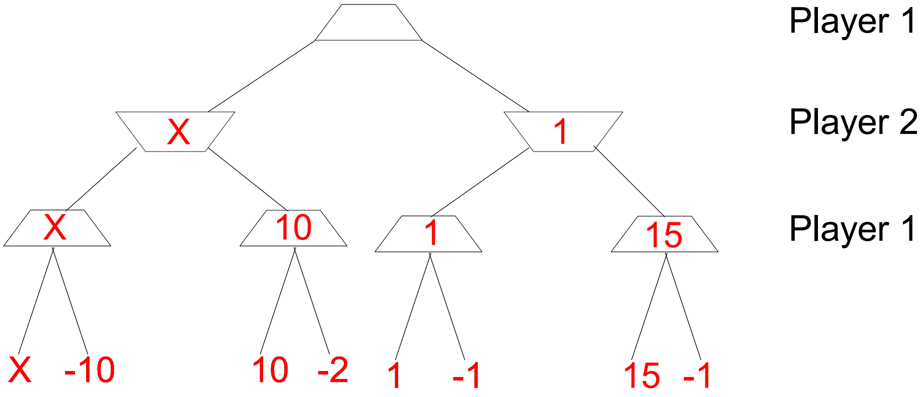


Player 1 should play Right for a payoff of 8.

Consider the following modified game tree, where one of the leaves has an unknown payoff x. Player 1 moves first, and attempts to maximize the value of the game.



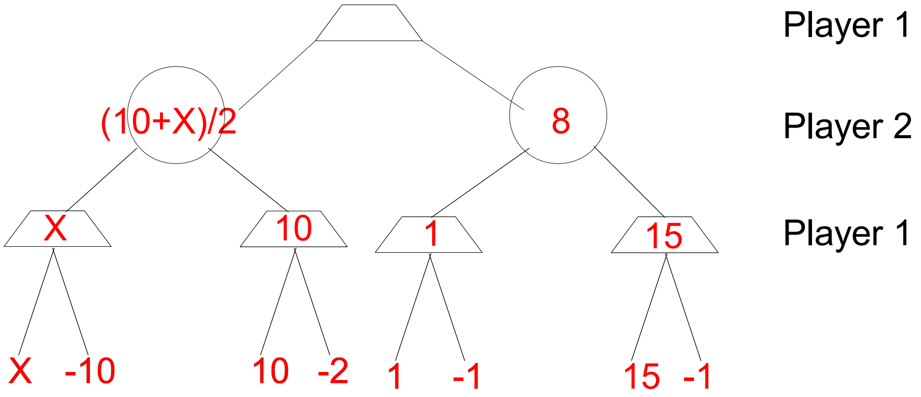
(d) Assume Player 2 is a minimizing agent (and Player 1 knows this). For what values of x does Player 1 choose the left action?



First, consider under which conditions the player will ultimately receive the x payout. The bottom left maximizing agent will choose the action leading to x if $x > -10$. The minimizing agent above it will choose the action leading to x as long as $x < 10$. The topmost maximizing agent will choose the action leading to x if $x > 1$.

So, for any $x > 1$, Player 1 will choose the left action (we also accepted $x \geq 1$).

(e) Assume Player 2 chooses actions at random (and Player 1 knows this). For what values of x does Player 1 choose the left action?



Running the expectimax calculation we find that the left branch is worth $(10 + x)/2$ while the right branch is worth 8. Calculation shows that the left branch has higher payoff if $x > 6$. ($x \geq 6$ also acceptable)

(f) For what values of x is the minimax value of the tree worth more than the expectimax value of the tree?

The minimax value of the tree can never exceed the expectimax value of the tree, because the only chance nodes are min nodes. The value of the min node is (weakly) less than the value of the corresponding chance node, so the value the max player receives at the root is (weakly) less under minimax than expectimax.

Common mistakes:

- If you assume the minimax value of the tree is x for $x > 1$ and the expectimax value of the tree is $(x+10)/2$ for $x > 6$ and solve the inequality, you get $x > 10$ as the critical value for x, corresponding to a minimax payoff of x. However, the minimax value can not exceed 10, or the min player will choose the branch with value 10, so you cannot get payoffs $x > 10$.
- If you calculate the value of the left branch under minimax rules and the value of the right branch under expectimax, you may get $x > 8$.