Written Assignment 2: Solution

Deadline: Oct 25th, 2023

Instruction: You may discuss these problems with classmates, but please complete the write-ups individually. (This applies to BOTH undergraduates and graduate students.) Remember the collaboration guidelines set forth in class: you may meet to discuss problems with classmates, but you may not take any written notes (or electronic notes, or photos, etc.) away from the meeting. Your answers must be **typewritten**, except for figures or diagrams, which may be hand-drawn. Please submit your answers (pdf format only) on **Canvas**.

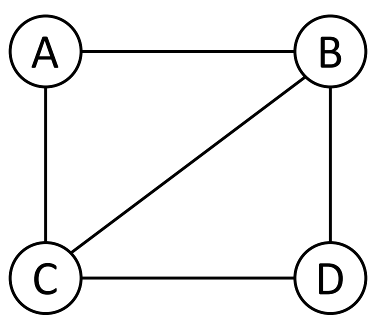
Q1. Solving CSPs (25 points)

In a combined 3^{rd} and 5^{th} grade class, students can be 8, 9, 10, and 11 years old. We are trying to solve for the ages of **A**nn, **B**ob, **C**laire, and **D**oug (abbreviations: **A**, **B**, **C**, **D**). Consider the following constraints:

- No students is older in years than Clair (but may be the same age)
- Bob is two years older than Ann
- Bob is younger in years than Doug.

Complete the following questions:

1. (5 pts) Draw the constraint graph.



Answer. Note: Two directed arcs in place of each undirected arc is also acceptable.

2. (3 pts) Suppose we are using the AC-3 algorithm for arc consistency. How many total arcs will be en-queued when the algorithm begins execution?
   Answer. 10. Remember that a relationship between two variables will result in two arcs (since arcs are directed.)

3. (9 pts) Assuming all ages $\{8, 9, 10, 11\}$ are possible for each student before running arc consistency, manually run arc consistency on **only** arc from **A** to **B**.

   (a) What values on A remain viable after this operation?
   Answer. 8, 9

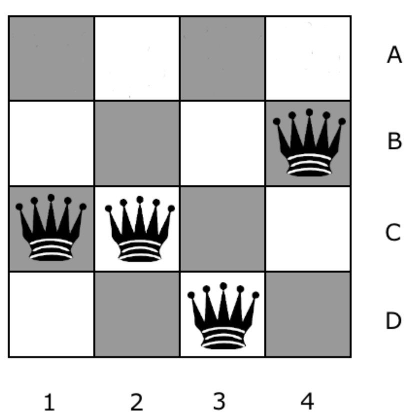
   (b) What values on B remain viable after this operation?
   Answer. 8, 9, 10, 11. Remember that enforcing the arc $A \to B$ should never change the viable domain of B.

   (c) Assuming there are no arcs left in the list of arcs to be processed, which arc(s) would be added to the queue for processing after this operation?
   Answer. $B \to A$, $C \to A$

4. (8 pts) Suppose we enforce arc consistency on all arcs. What ages remain in each person's domain?
   Answer. A: 8, B: 10, C: 11, D: 11. If we enforce the arcs $B \to A$, $B \to D$, $D \to B$, $A \to B$, $C \to D$, then we can quickly find the only satisfying solution.

Q2. 4-Queens (12 points)

The min-conflicts algorithm attempts to solve CSPs iteratively. It starts by assigning some value to each of the variables, ignoring the constraints when doing so. Then, while at least one constraint is violated, it repeats the following: (1) randomly choose a variable that is currenly violating a constraint, (2) assign to it the value in its domain such that after the assignment the total number of constraints violated is minimized (among all possible selections of values in its domain).



In this question, you are asked to execute the min-conflicts algorithm on a simple problem: the 4-queens problem in the figure shown below. Each queen is dedicated to its own column (i.e. we have variables Q_1, Q_2, Q_3, and Q_4 and the domain for each one of them is $\{A, B, C, D\}$). In the configuration shown below, we have $Q_1 = C$, $Q_2 = C$, $Q_3 = D$, $Q_4 = B$. Two queens are in conflict if they share the same row, diagonal, or column (though in this setting, they can never share the same column).

You will execute min-conflicts for this problem three times, starting with the state shown in the figure above. When selecting a variable to reassign, min-conflicts chooses a conflicted variable at random. For this problem, assume that your random number generator always chooses the leftmost conflicted queen. When moving a queen, move it to the square in its column that leads to the fewest conflicts with other queens. If there are ties, choose the topmost square among them.

1. Starting with the queens in the configuration shown in the above figure, which queen will be moved, and where will it be moved to?
   Answer. Queen 1 and position A. Queens 1,2, and 3 are conflicted, so according to the specification, the leftmost queen is selected: Queen 1. It is moved to position A, because there are no conflicts with position A.

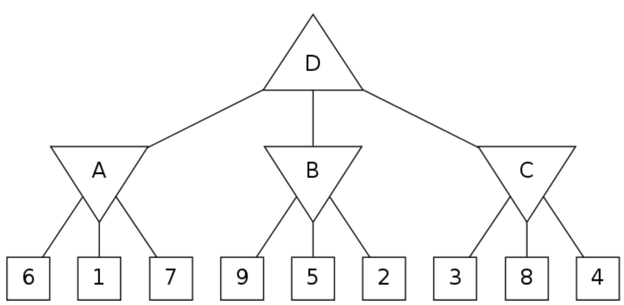
2. Continuing off of Part 1, which queen will be moved, and where will it be moved to?
   Answer. Queen 2 and position A. Queens 2 and 3 are conflicted, so Queen 2 is selected. Positions A and C have one conflict, so the topmost is selected: position A.

3. Continuing off of Part 2, which queen will be moved, and where will it be moved to?
   Answer. Queen 1 and position C. Now Queens 1 and 2 are conflicted, so Queen 1 is chosen again. This time position C has no conflicts, so it is moved back to position C.

Q3. Minimax and Expectimax (15 points)

Q3.1. Minimax (7.5 points)

Consider the zero-sum game tree shown below. Triangles that point up, such as at the top node (root), represent choices for the maximizing player; triangles that point down represent choices for the minimizing player. Outcome values for the maximizing player are listed for each leaf node, represented by the values in squares at the bottom of the tree. Assuming both players act optimally, carry out the minimax search algorithm. Enter the values for the letter nodes.



Answer. A: 1, B: 2, C: 3, D: 3

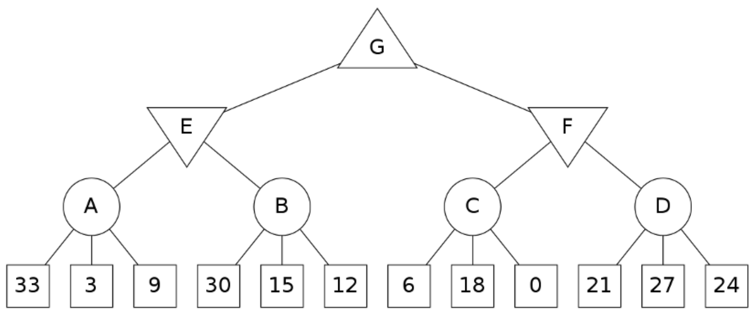
Explanation. Maximizing nodes choose the highest value from their children, and minimizing node take on the lowest value from among their children.

A chooses 1, B chooses 2, and C chooses 3.

D then chooses 3 from among A, B, and C.

Q3.2. Expectimax (7.5 points)

Consider the game tree shown below. As in the previous problem, triangles that point up, such as the top node (root), represent choices for the maximizing player; triangles that point down represent choices for the minimizing player. The circular nodes represent chance nodes in which each of the possible actions may be taken with equal probability. The square nodes at the bottom represent leaf nodes. Assuming both players act optimally, carry out the expectiminimax search algorithm. Enter the values for the letter nodes.



Answer. A: 15, B: 19, C:8, D:24, E:15, F: 8, G: 15.

Explanation. The value for each circular node is equal to the expectation of the values of its children, which is found by adding up each of the child values and dividing by the number of children.

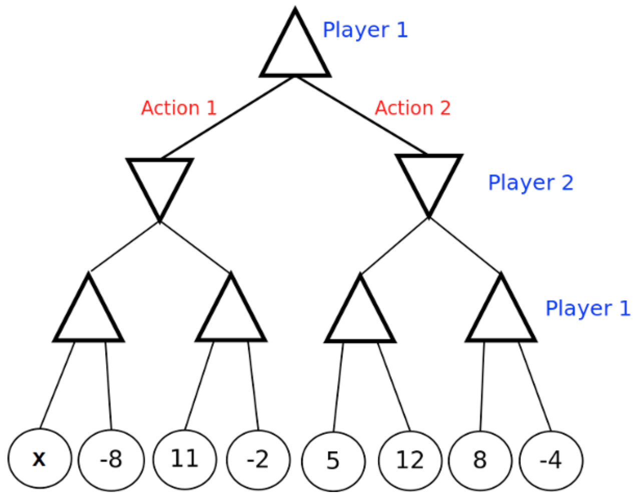
E and F take the minimums from their respective children.

G takes 15, the max of E and F

Q4. Unknown Leaf Value (28 points)

Consider the following 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. Each of the next 3 questions asks you to write a constraint on x specifying the set of values it can take. Please specify your answer in one of the following forms:

- Write All if x can take on all values
- Write None if x has no possible values
- Use an inequality in the form $x < \{value\}$, $x > \{value\}$, or $\{value1\} < x < \{value2\}$ to specify an interval of values. As an example, if you think x can take on all values larger than 16, you should enter $x > 16$.



Q4.1. Assume Player 2 is a minimizing agent (and Player 1 knows this). For what values of x is Player 1 guaranteed to choose Action 1 for their first move?

Answer. $x > 8$

Explanation. Depending on the value of x, there can be three different values for taking Action 1:

If $x < -8$, then Action 1 results in -8;
If $-8 \leq x \leq 11$ then Action 1 results in x;
If $x > 11$ then Action 1 results in 11.
Action 2 always results in a utility of 8
Hence Action 1 is optimal for Player 1 if $x > 8$

Q4.2. Assume Player 2 chooses actions at random with each action having equal probability (and Player 1 knows this). For what values of x is Player 1 guaranteed to choose Action 1?

Answer. $x > 9$

Explanation. Action 2 gives Player 1 a utility of 10, so the average of x and 11 must be greater than 10.

$(x + 11)/2 > 10 \implies x > 9$

Q4.3. Denote the minimax value of the tree as the value of the root when Player 1 is the maximizer and Player 2 is the minimizer. Denote the expectimax value of the tree as the value of the root when Player 1 is the maximizer and Player 2 chooses actions at random (with equal probability). For what values of x is the minimax value of the tree worth more than the expectimax value of the tree?

Answer. None.

Explanation. To satisfy this, you would need x s.t. $\max(\min(x, 11), 8) > \max((x + 11)/2, 10)$

For $x \leq 8$, minimax has value 8 and expectimax has value 10 since $(x+11)/2 < 10$. And $8 < 10$

For $8 < x \leq 9$, minimax has value x and expectimax has value 10 since $(x + 11)/2 \leq 10$. And $x < 10$.

For $9 < x \leq 11$, minimax has value x and expectimax has value $(x+11)/2$. And $x \leq (x+11)/2$

For $11 < x$, minimax has value 11 and expectimax has value $(x + 11)/2$. And $11 < (x + 11)/2$

Q4.4. Is it possible to have a game, where the minimax value is strictly larger than the expectimax value?

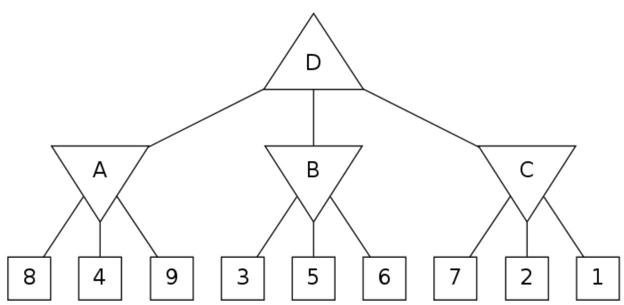
Answer. No

Explanation. The minimax value can never be strictly greater than the expectimax value for the same tree because in minimax Player 2 always chooses the worst possible move for Player 1, while in expectimax, those same nodes average that value with other higher values. Thus, the utility at a node under expectimax is always at least as high as the utility of the same node under minimax.

Q5. Alpha-Beta Pruning (20 points)

Consider the game tree shown below. Triangles that point up, such as at the top node (root), represent choices for the maximizing player; triangles that point down represent choices for the minimizing player. Assuming both players act optimally, use alpha-beta pruning to find the value of the root node. The search goes from left to right; when choosing which child to visit first, choose the left-most unvisited child.

Hint: Note that the value of a node where pruning occurs is not necessarily the maximum or minimum (depending on which node) of its children. When you prune on conditions, $V > \beta$ or $V < \alpha$, assume that the value of the node is V.



Q5.1. (10 points) Enter the values of the labeled nodes.

Answer. A: 4, B: 3, C: 2, D: 4

Explanation. See below.

Q5.2. (10 points) Select the leaf nodes that don't get visited due to pruning.

Answer. Boxes 5, 6, and 1.

Explanation.

- A: $\alpha = -\infty$ and $\beta = \infty$. Because $\lambda = -\infty$, there will be no pruning. Intuitively, this means that any value that the minimizer finds might be used by the maximizer.
- B: $\alpha = 4$ and $\beta = \infty$. The first leaf value, 3, is less than 4, so the remaining children can be pruned and B gets value 3. Intuitively, B will never take a value greater than 3, and D will never select a value less than 4, so you know that D will never select the middle action.
- C: $\alpha = 4$ and $\beta = \infty$. The first leaf value, 7, is greater than 4, so the other children must be checked. The second leaf value, 2, is less than 4, so the remaining child can be pruned and C gets value 2.
- D: Because $\beta = \infty$, none of D's children can be pruned, which is always true for the root. D then takes the max of its children, and has value 4 (which, because none of its children were pruned, is identical to its value when running minimax without pruning).