Section Handout 3 Solutions

Q1. CSP: Air Traffic Control

We have five planes: A, B, C, D, and E and two runways: international and domestic. We would like to schedule a time slot and runway for each aircraft to **either** land or take off. We have four time slots: $\{1, 2, 3, 4\}$ for each runway, during which we can schedule a landing or take off of a plane. We must find an assignment that meets the following constraints:

- Plane B has lost an engine and must land in time slot 1.

- Plane D can only arrive at the airport to land during or after time slot 3.

- Plane A is running low on fuel but can last until at most time slot 2.

- Plane D must land before plane C takes off, because some passengers must transfer from D to C.

- No two aircrafts can reserve the same time slot for the same runway.

(a) Complete the formulation of this problem as a CSP in terms of variables, domains, and constraints (both unary and binary). Constraints should be expressed implicitly using mathematical or logical notation rather than with words.

   Variables: A, B, C, D, E for each plane.

   Domains: a tuple *(runway type, time slot)* for runway type $\in \{international, domestic\}$ and time slot $\in \{1, 2, 3, 4\}$.

   Constraints:

$$B[1] = 1$$

$$D[1] \geq 3$$

$$A[1] \leq 2$$

$$D[1] < C[1]$$

$$A \neq B \neq C \neq D \neq E$$

(b) For the following subparts, we add the following two constraints:

   - Planes A, B, and C cater to international flights and can only use the international runway.
   - Planes D and E cater to domestic flights and can only use the domestic runway.

   (i) With the addition of the two constraints above, we completely reformulate the CSP. You are given the variables and domains of the new formulation. Complete the constraint graph for this problem given the original constraints and the two added ones.

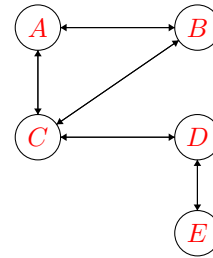   **Variables:** A, B, C, D, E for each plane.

   Domains: $\{1, 2, 3, 4\}$

   Explanation of Constraint Graph:
   We can now encode the runway information into the identity of the variable, since each runway has more than enough time slots for the planes

Constraint Graph:

(ii) What are the domains of the variables after enforcing arc-consistency? Begin by enforcing unary constraints. (Cross out values that are no longer in the domain.)

| A	1	2	3	4
B	1	2	3	4
C	1	2	3	4
D	1	2	3	4
E	1	2	3	4

2 Games

(a) 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. Assuming both players act optimally, fill in the minimax value of each node.
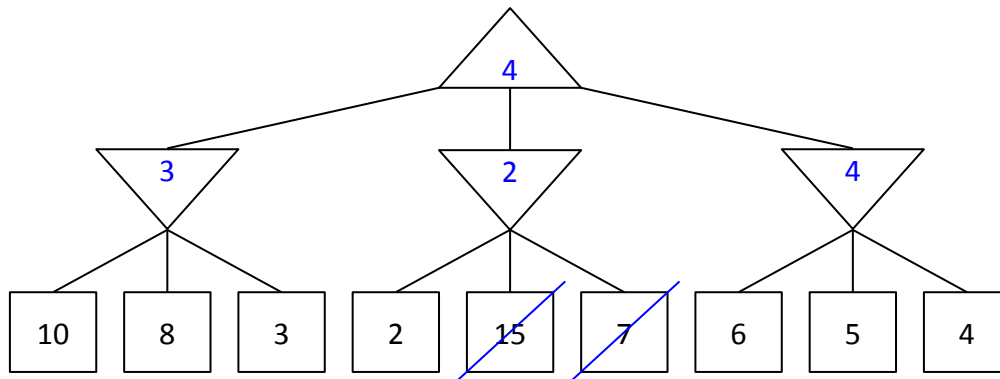
(b) Which nodes can be pruned from the game tree above through alpha-beta pruning? If no nodes can be pruned, explain why not. Assume the search goes from left to right; when choosing which child to visit first, choose the left-most unvisited child.

(c) (optional) Again, consider the same zero-sum game tree, except that now, instead of a minimizing player, we have a chance node that will select one of the three values uniformly at random. Fill in the expectimax value of each node. The game tree is redrawn below for your convenience.
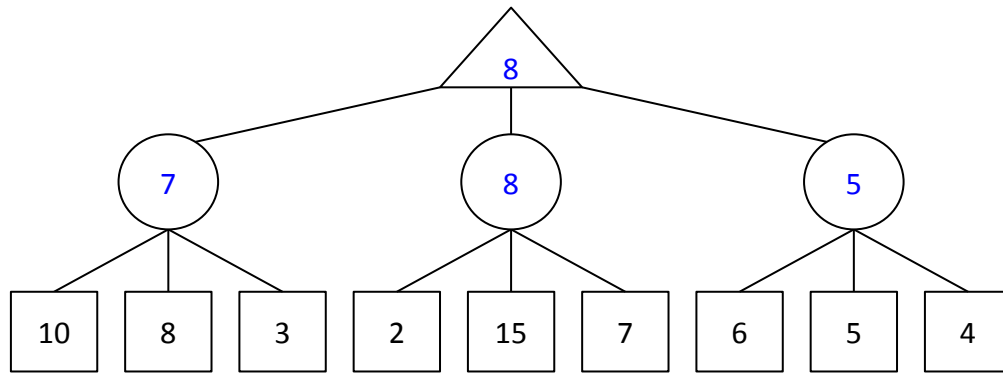
(d) (optional) Which nodes can be pruned from the game tree above through alpha-beta pruning? If no nodes can be pruned, explain why not.   No nodes can be pruned. There will always be the possibility that an as-yet-unvisited leaf of the current parent chance node will have a very high value, which increases the overall average value for that chance node. For example, when we see that leaf 4 has a value of 2, which is much less than the value of the left chance node, 7, at this point we cannot make any assumptions about how the value of the middle chance node will ultimately be more or less in value than the left chance node. As it turns out, the leaf 5 has a value of 15, which brings the expected value of the middle chance node to 8, which is greater than the value of the left chance node. In the case where there is an upper bound to the value of a leaf node, there is a possibility of pruning: suppose that an upper bound of $+10$ applies only to the children of the rightmost chance node. In this case, after seeing that leaf 7 has a value of 6 and leaf 8 has a value of 5, the best possible value that the rightmost chance node can take on is $\frac{6+5+10}{3} = 7$, which is less than 8, the value of the middle chance node. Therefore, it is possible to prune leaf 9 in this case.

3 Nonzero-sum Games

1. Let's look at a non-zero-sum version of a game. In this formulation, player A's utility will be represented as the first of the two leaf numbers, and player B's utility will be represented as the second of the two leaf numbers. Fill in this non-zero game tree assuming each player is acting optimally.
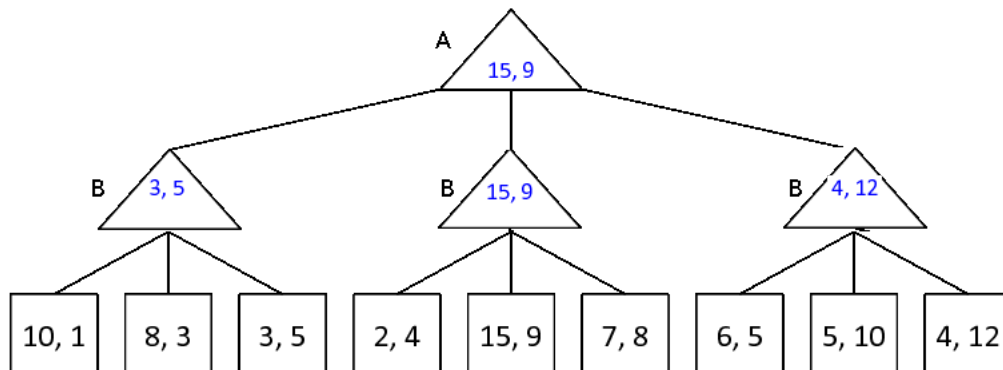
2. Which nodes can be pruned from the game tree above through alpha-beta pruning? If no nodes can be

pruned, explain why not. No nodes can be pruned. Because this game is non-zero-sum, there can exist a leaf node anywhere in the tree that is good for both player A and player B.