CMSC 351 Summer 2025 Homework 10

Due Monday 18 August 2025 by 23:59 EST on Gradescope.

Directions:

- Homework must be done on printouts of these sheets and then scanned properly, or via Latex, or by downloading, writing on the PDF, and uploading. If you use Latex please do not change the Latex formatting.
- Do not use your own blank paper!
- The reason for this is that Gradescope will be following this template to locate the answers to the problems so if your answers are organized differently they will not be recognized.
- Tagging is automatic, you will not be able to manually tag.

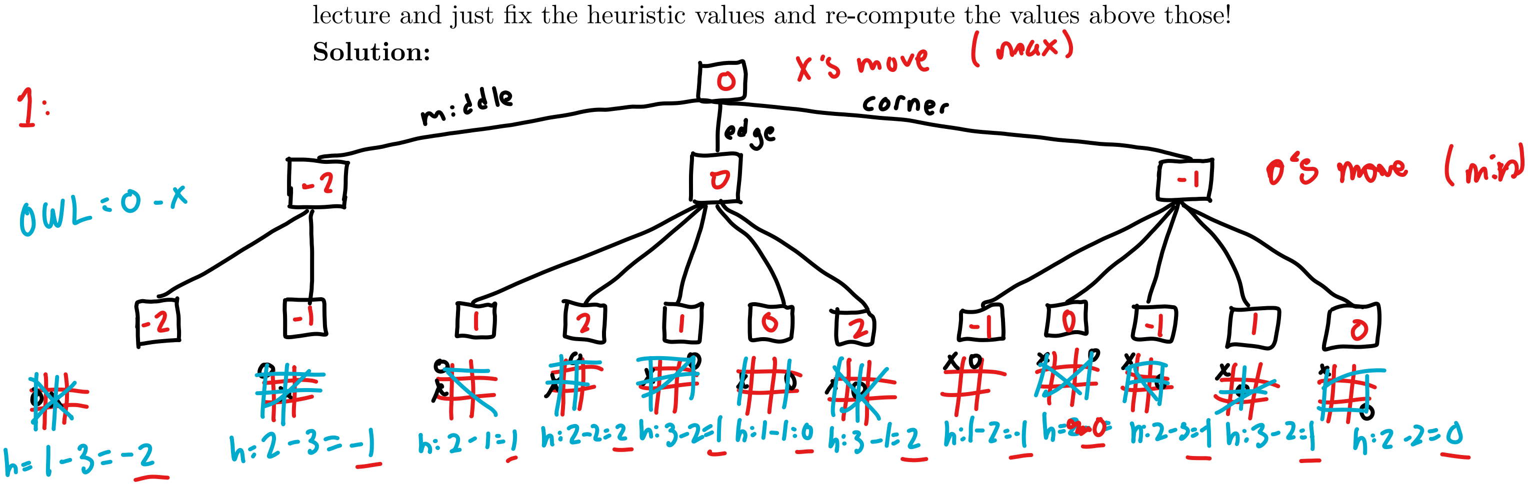
---

1. In the misère version of tic-tac-toe a player wins if their opponent gets three in a row; the basic idea being you want to force your opponent to get three in a row so you win. Assuming X goes first it then makes sense to define the heuristic to be:

$$h = (\#\text{OWLs for } O) - (\#\text{OWLs for } X)$$

(a) Draw the Minimax tree for the misère version for two moves. Hint: Use the one from the lecture and just fix the heuristic values and re-compute the values above those! [15 pts]

Solution:



(b) What is the best move for X? [5 pts]

Solution:

Pick any edge first move (b/c symmetry)
and allow O to pick the opposite edge

2. Below is the board for a simple game, with two tokens, `A` and `B`:

A					B
1	2	3	4	5	6

Player `A` moves first, and the two players take turns moving. Player `A` can move either one, two, or three squares right while player `B` can move either one, two, or three squares left. A player may not land on the square their opponent occupies but may jump their opponent and if they do this then their opponent must do nothing for their next move. The winner is the player who reaches the opposite end first.

For any position define the heuristic function:

$$h = [\text{Number of squares A has moved}] - [\text{Number of squares B has moved}]$$

(a) Draw a game tree for the first three moves (`A` moves, `B` moves, `A` moves). This tree should show the actual board arrangements! [20 pts]

Solution:

A moves first

1 move ahead

_ A _ _ _ B A1 h=1

_ _ A _ _ B A2 h=2

_ _ _ A _ B A3 h=3

2 moves ahead

From A1, B has 3 moves

_ A _ _ B _ B1 h=0

_ A _ B _ _ B2 h=-1

_ A B _ _ _ B3 h=-2

From A2, B has 2

_ _ A _ B _ B4 h=1

_ _ A B _ _ B5 h=0

From A3, B has 2

_ _ _ A B _ B6 h=2

_ _ B A _ _ B7 h=0

3 moves ahead:

from B1, A has 2 moves

_ _ A _ B _ h=1

_ _ _ A B _ h=2

from B2, A: 2

_ _ A B _ _ h=0

_ _ _ B A _ h=2

from B3, A:

_ B A _ _ _ h=0

_ B _ A _ _ h=1

from B4: A: 2

_ _ _ A B _ h=2

_ _ _ _ B A h=4

from B5, A: 2

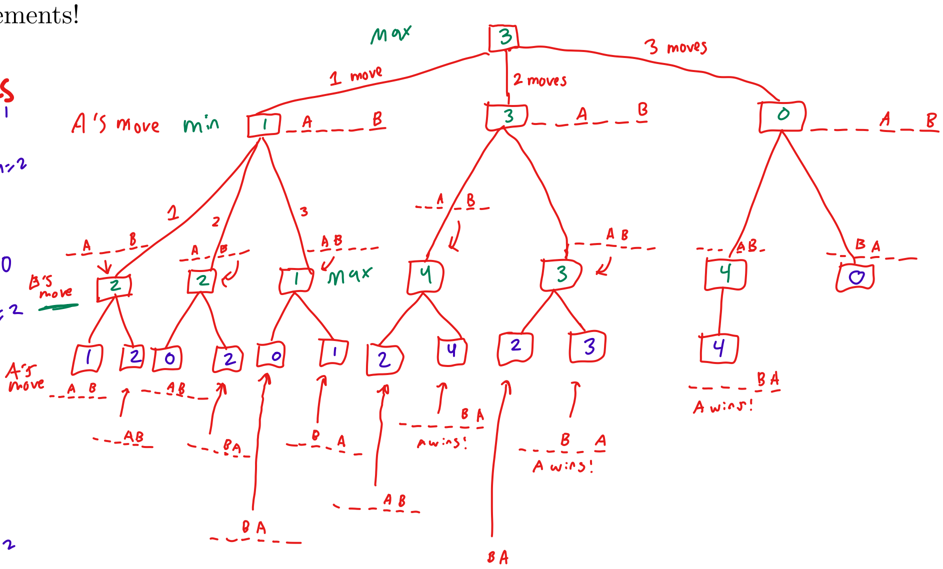
_ _ _ B A _ h=2

_ _ _ B _ A h=3

from B6: A: 1

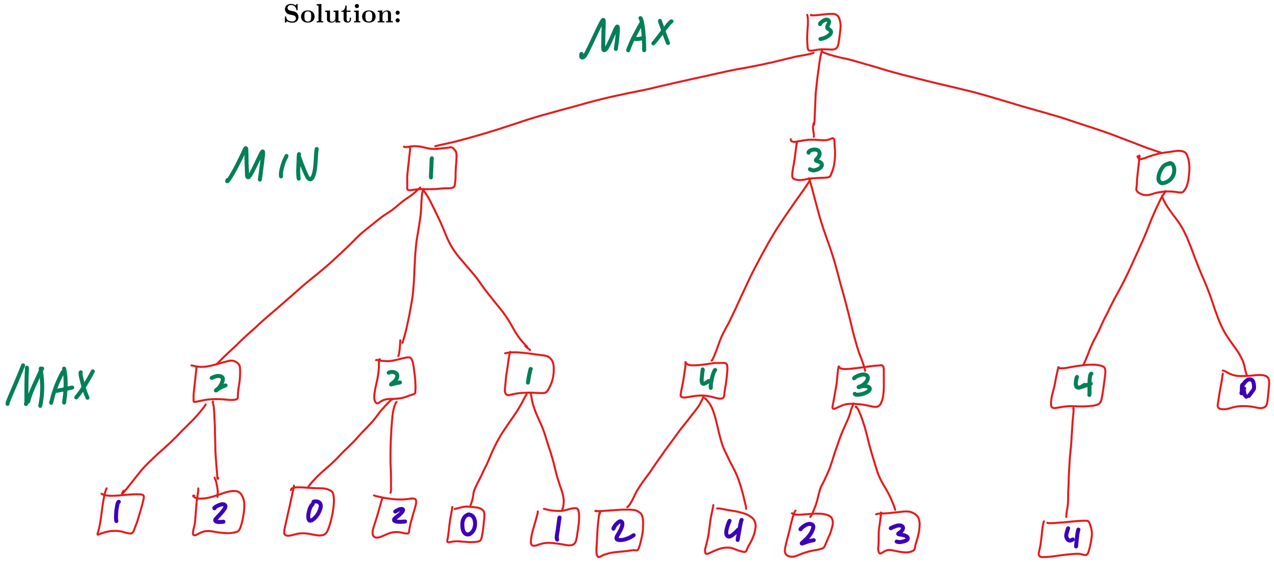
_ _ _ _ B A h=4

from B7: A: 0 moves h=0



(b) Draw a second tree which contains the heuristic values for the leaves and then fill in the remaining values according to the Minimax algorithm. [15 pts]

Solution:



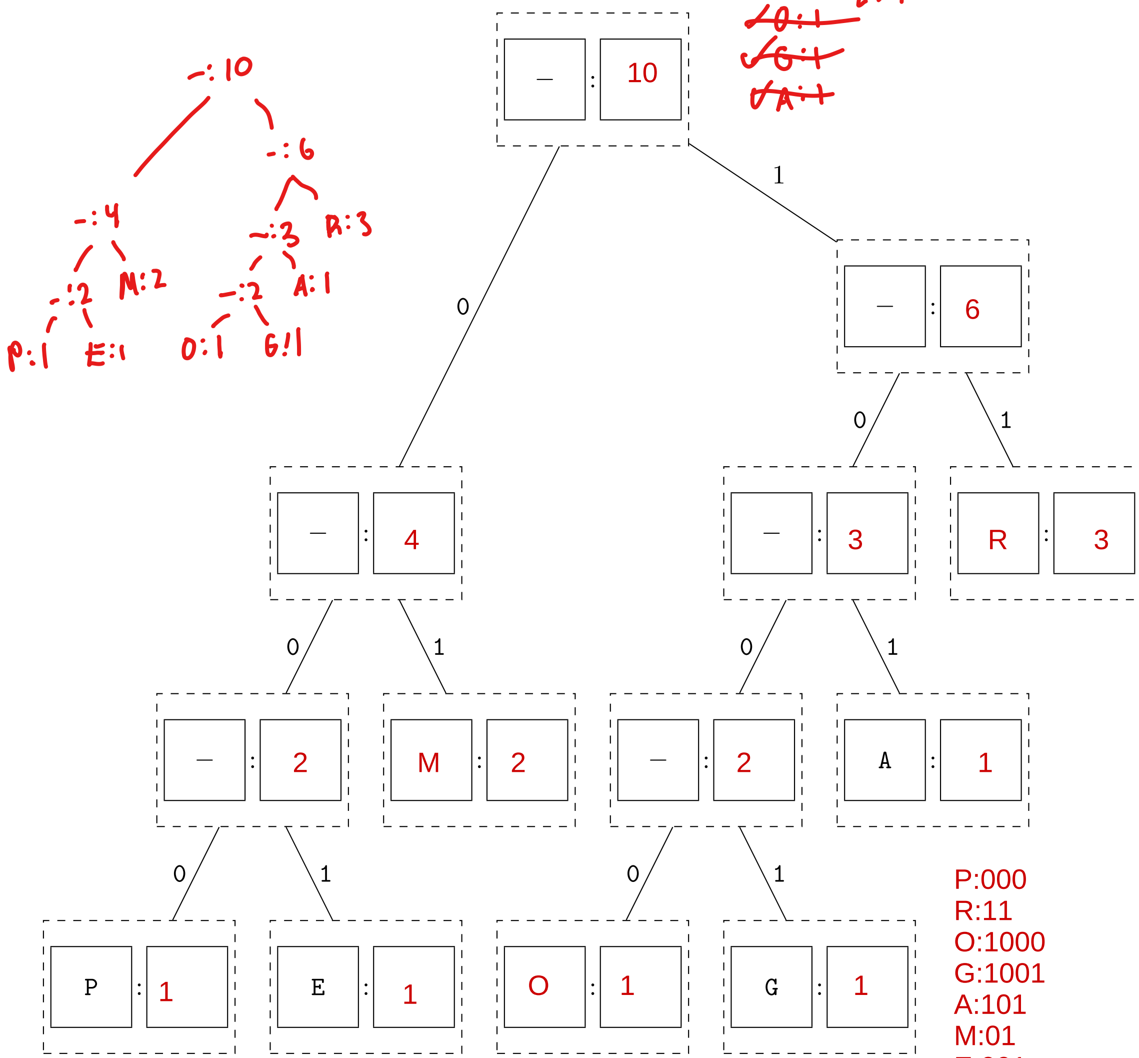
(c) What should `A` do for their first move? [5 pts]

Solution:

According to this minimax tree, A should move 2 spaces on their first move.

3. The tree below is the result of running Huffman's algorithm encoding PROGRAMMER. Fill in the all the missing data. There is only one way to do this. [16 pts]

~~P:1~~ M:2
R:3 ~~E:1~~
~~O:1~~
~~G:1~~
~~A:1~~



P:000
R:11
O:1000
G:1001
A:101
M:01
E:001

What is the encoding of PROGRAMMER? [4 pts]

Encoding of PROGRAMMER	000 11 1000 1001 11 101 01 01 001 11

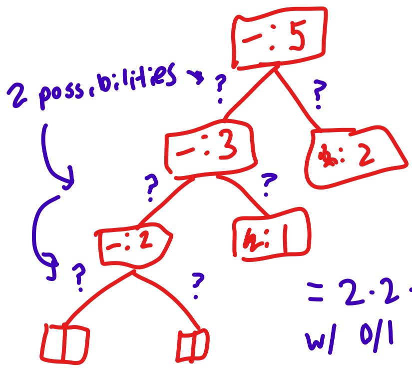
P R O G R A M M E R

4. How many different Huffman encodings of the word DREAD are there? You should take into account all possible options during tree construction and all possible assignments of 0 and 1 branch labels. Explain! [20 pts]

Solution:

D: 2
R: 1
E: 1
A: 1

3 choices at beginning for the smallest 2 nodes (R,E,A)
after that, 2 of those will become 1 and the count turns to 2
the set of counts becomes {2,2,1}
from here you can choose either of the 2 counts to be the "smallest" along with the 1 so that is another 2 choices.
this turns the set into {3,2} and from here there is only one outcome.
multiplying the outcomes so far: 3 * 2 = 6 outcomes when building the tree.



If you take into account the assignments of 0 and 1, there are 8 more possibilities because the resulting tree has 3 parent nodes (nodes with 2 children) and at each of those nodes you can assign 0/1 to be left or right so there is 2 possibilities at each level * 3 parent nodes = 8 outcomes when assigning 0/1.

$= 2 \cdot 2 \cdot 2 = 8$ possibilities w/ 0/1 assignment alone.

8 * 6 = 48
there are 48 possible Huffman encodings "DREAD"