# CS 461 – ARTIFICIAL INTELLIGENCE

## HOMEWORK #3 (5% or 10 points)

Assigned: Wed Oct 28, 2020

Due: Wed Nov 11, 2020 **2:00 pm**

Do not forget to indicate the group members submitting the homework (at most 5 names). Submit your homework according to your TAs' guidelines.

Feel free to use any programming language as long as any of you can, if requested, give a demo using your notebook computer.

The usual late submission policy applies.

## PROBLEM

Consider the following Minimax Tree. If we evaluate children from left-to-right using alpha-beta pruning with a given set of values for nodes A through I, some of the nodes may be pruned. That is, we may never need to evaluate the node because of what we have seen in prior search.
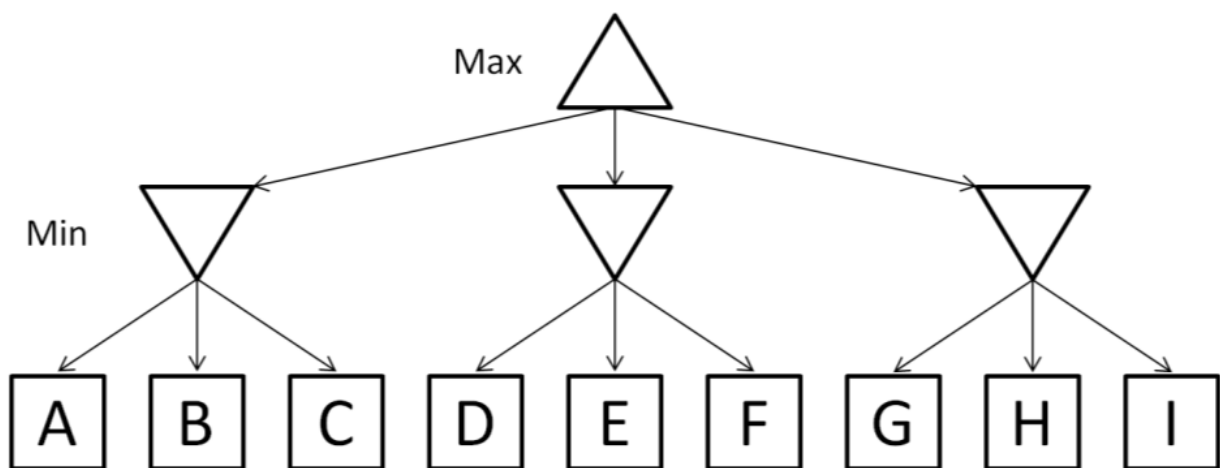
In this homework, you'll write two programs, specifically tailored for this particular tree. (In other words, your programs will work only for trees having the precise format shown above.)

- The first program will be called *MINIMAX* and will perform straight minimaxing. It'll take as input the values of the nodes A through I and will output the move of the Max player as one of "L" (left), "M" (middle), or "R" (right).
- The second program will be called *ALPHABETA* and will perform alpha-beta pruning. It'll take as input the values of the nodes A through I and will output the move of the Max player as one of "L" (left), "M" (middle), or "R" (right). <u>Additionally</u>, it'll output, in alphabetical order and separated by blanks, the names of the nodes (if any) that are pruned.

## TEST DATA FOR *MINIMAX*

1. (1 point) Tree #1: A=5 B=3 C=1 D=2 E=5 F=4 G=1 H=3 I=3
2. (2 points) Tree #2: ...

   The node values for this tree are not known at this stage; they will be manually entered by your TA when your submission is being evaluated. Thus, your program should be prepared to input 9 values (domain: $Z$), separated by blanks, from the keyboard.

## TEST DATA FOR *ALPHABETA*

1. (1 point) Tree #1: A=5 B=3 C=1 D=2 E=5 F=4 G=1 H=3 I=3
2. (1 point) Tree #2: A=5 B=2 C=2 D=5 E=1 F=3 G=2 H=4 I=2
3. (1 point) Tree #3: A=1 B=3 C=4 D=1 E=4 F=1 G=3 H=5 I=3
4. (4 points) Tree #4: ...

   The node values for this tree are not known at this stage; they will be manually entered by your TA when your submission is being evaluated. Thus, your program should be prepared to input 9 values (domain: $Z$), separated by blanks, from the keyboard.

Both programs should have a simple control for 'single stepping' (tracing your code) so that one can inspect the intermediate stages of the problem-solving process in an incremental fashion.

CAVEAT: Even a single missing or erroneous output in a test is sufficient to nullify that test. In other words, precise correct answers are required for each of the 6 trees above.