

# CS1571 – Introduction to Artificial Intelligence

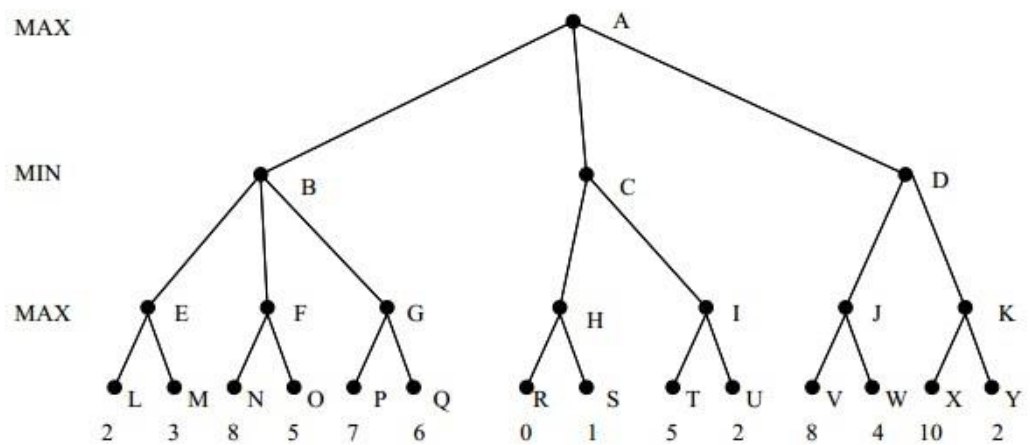
## Assignment 4

Name: You Zhou ([yoz13@pitt.edu](mailto:yoz13@pitt.edu))

Problem\_1:

(a)

A	8
B	3
C	1
D	8
E	3
F	8
G	7
H	1
I	5
J	8
K	10



Best move:  $V \rightarrow J \rightarrow D \rightarrow A$

(b) Left  $\rightarrow$  Right: OQTIUY.

(c) Right  $\rightarrow$  Left: SHROFNMEL.

Problem\_2:

(a)

	PlayerA	PlayerB
Round0	Draw	Draw
Round1	Draw	Draw
Round2	Draw	Draw
Round3	Draw	Draw
Round4	Draw	Draw
Round5	Draw	Draw
Round6	Draw	Draw
Round7	Won	Lose
Round8	Draw	Draw
Round9	Won	Lose

(b)

	MyPlayerA	PlayerB
Round0	Won	Lose
Round1	Draw	Draw
Round2	Won	Lose
Round3	Lose	Won
Round4	Lose	Won
Round5	Won	Lose
Round6	Draw	Draw
Round7	Won	Lose
Round8	Draw	Draw
Round9	Won	Lose

	PlayerA	MyPlayerB
Round0	Draw	Draw
Round1	Draw	Draw
Round2	Won	Lose
Round3	Lose	Won
Round4	Lose	Won
Round5	Draw	Draw
Round6	Draw	Draw
Round7	Won	Lose
Round8	Draw	Draw
Round9	Draw	Draw

Analysis:

When my heuristics moved first, it won 3 times, lost 2 times, and drew 5 times. When the original heuristics moved first, it won 2 times, lost 2 times, and drew 6 times. Since the number of trials is too small, we cannot conclude that one of these two heuristics is better than the other one.

When choosing configurations, I let the number of the same symbols possible to combine a winning configuration be the dominating factor, some of them are: “\_XXXX\_,” “XX\_XX,” “X\_XXX\_,” “\_XX\_XX,” “\_XXX\_X,” “\_OOO\_,” “\_OO\_O\_,” “\_O\_O\_.” Quite intuitively, the more number of this kind of symbols a configuration contains, the more likely it could win.

The original heuristics doesn't consider the potential of configurations containing a certain number of symbols but blocked by the opponent on one side, such as “\_OOOOX,” “XOO\_OO,” “\_\_XXXO,” “X\_OO\_O.” Even though they are less likely to achieve winning configuration than those I listed in the above paragraph are, but they need to be assigned a heuristic value for improving the precision.

Putting the strategies so far together: The number of the same symbols possible to win is the most crucial factor. Having the same number, whether or not the configuration is blocked by the opponent is the second crucial factor. Abstractly saying:

“\_XXXX\_” > “OXXXX\_” and “\_OO\_O\_” > “XOO\_O\_”.

As the number of the same symbols decreases to 2 (“\_XX\_,” “XX\_.”), we should consider all its permutations as many as possible. Hence, “XX\_,” “X\_X,” “\_XX\_,” “\_X\_X\_,” and all others should be assigned a heuristic value, so we obtain a higher degree of variability during the early stage of the game.

The key idea of assigning heuristic value is to assign a relatively small value to configurations containing only 1 or 2 same symbols. For example, we assign 800 to configuration “\_XXXX\_,” but we only assign 5 to “\_X\_X\_.” The reason for it is, as the game board becomes congested, the number of those less significant configurations will increase dramatically, possibly diverting computer from choosing a superior one. When adjusting my program at the beginning, I noticed that the computer didn't assign another symbol to either side of “\_XXXX\_,” instead, it filled the middle position of “X\_X.” This is because the heuristic overestimate the potential of it. To avoid that, we have to put configurations on the level of 3 and 4 into the dominating position so that, once it appears, it will filter all other less significant configurations.