CS-171, Intro to A.I., Fall Quarter, 2018—Quiz # 3—20 minutes

NAME:______________________________________ UCI NetID: ____________________

YOUR ID#: _________ ID# TO RIGHT: ________ ID# TO LEFT: ___________ ROW:_____ SEAT:_____

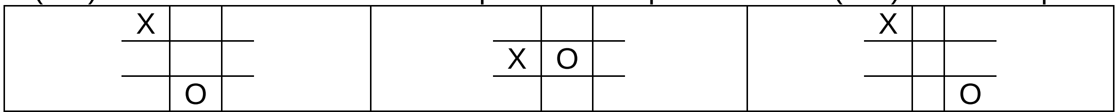
1. (40 pts total, 4 pts each) ADVERSARIAL (GAME) SEARCH CONCEPTS.
For each of the following terms on the left, write in the letter corresponding to the best answer or the correct definition on the right.

See Chapter 5.

D	Game Strategy	A	Approximates the value of a game state (i.e., of a game position)
H	Cut-off Test	B	In all game instances, total pay-off summed over all players is a constant
E	Alpha-Beta Pruning	C	Tree where nodes are game states and edges are game moves
G	Weighted Linear Function	D	Function that specifies a player's move in every possible game state
J	Terminal Test	E	Returns same move as MiniMax, but may prune more branches
I	Monte Carlo Tree Search	F	Optimal strategy for 2-player zero-sum games of perfect information, but impractical given limited time to make each move
C	Game Tree	G	Vector dot product of a weight vector and a state feature vector
A	Heuristic Evaluation Function	H	Function that decides when to stop exploring this search branch
B	Zero-sum Game	I	Play out many games randomly, and use the results as a score
F	MiniMax Algorithm	J	Function that says when the game is over

****** TURN PAGE OVER AND CONTINUE ON THE OTHER SIDE ******

2. (60 pts total) GAME SEARCH WITH TIC-TAC-TOE AND WIN-PATHS HEURISTIC FUNCTION.
This problem asks about MiniMax Search and Alpha-Beta pruning in Tic-Tac-Toe with the Win-paths static heuristic evaluation function. Recall that the Win-paths heuristic function counts the number of possible win-paths for MAX (= X) and subtracts the number of possible win-paths for MIN (= O). For example:

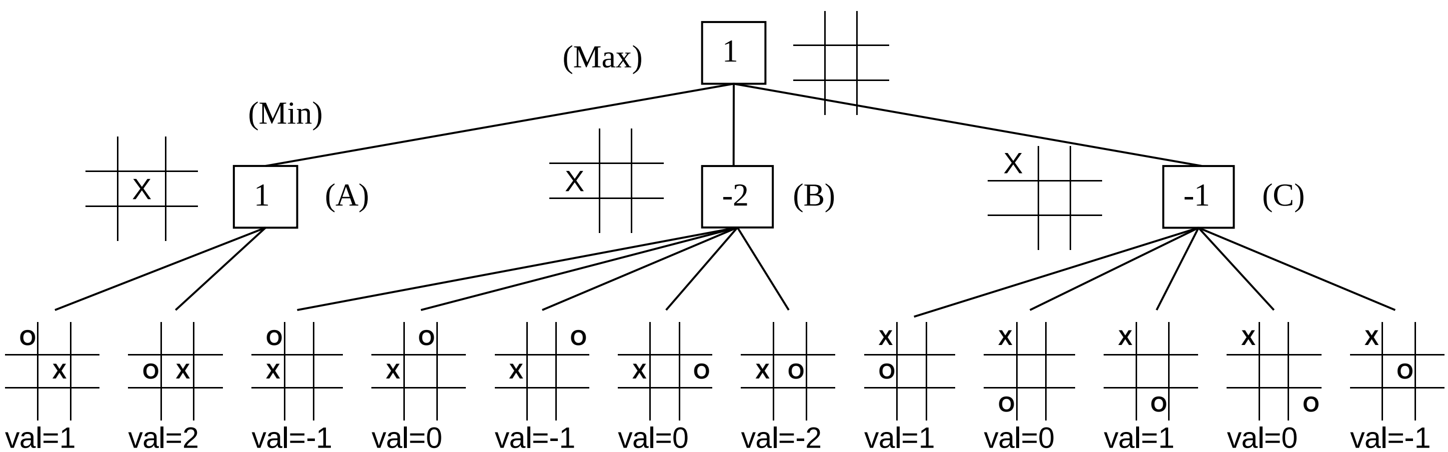


MAX (= X) # win paths=6 MIN (= O) # win paths=5 Heuristic value = 1 (= 6-5)	MAX (= X) # win paths=4 MIN (= O) # win paths=6 Heuristic value = -2 = (4-6)	MAX (= X) # win paths=5 MIN (= O) # win paths=5 Heuristic value = 0 (= 5-5)

2.a. (24 pts total, 4 pts each blank branch node [4 in part 1] or answer space [1 each in parts 2&3])
In the game tree below it is **Max**'s (= X's) turn to move. At each leaf node is the estimated score of that resulting position as returned by the Win-path heuristic static evaluator (written below as "val=*n*").
(1) Perform Mini-Max search and label each branch node with its return value (4 branch nodes).

(2) What is Max's best move (A, B, or C)? ______A___________

(3) What value does Max expect to get? ________1____________



2.b. (36 pts total, 3 pts each leaf node)
In the game tree below it is **Max**'s (= X's) turn to move (this is the same game tree as in problem 2.a above). At each leaf node is the estimated score of that resulting position as returned by the Win-path heuristic static evaluator (as "val=*n*"). You do <u>not</u> need to indicate the branch node return values again.

Cross out each leaf value that would be pruned by Alpha-Beta Pruning.