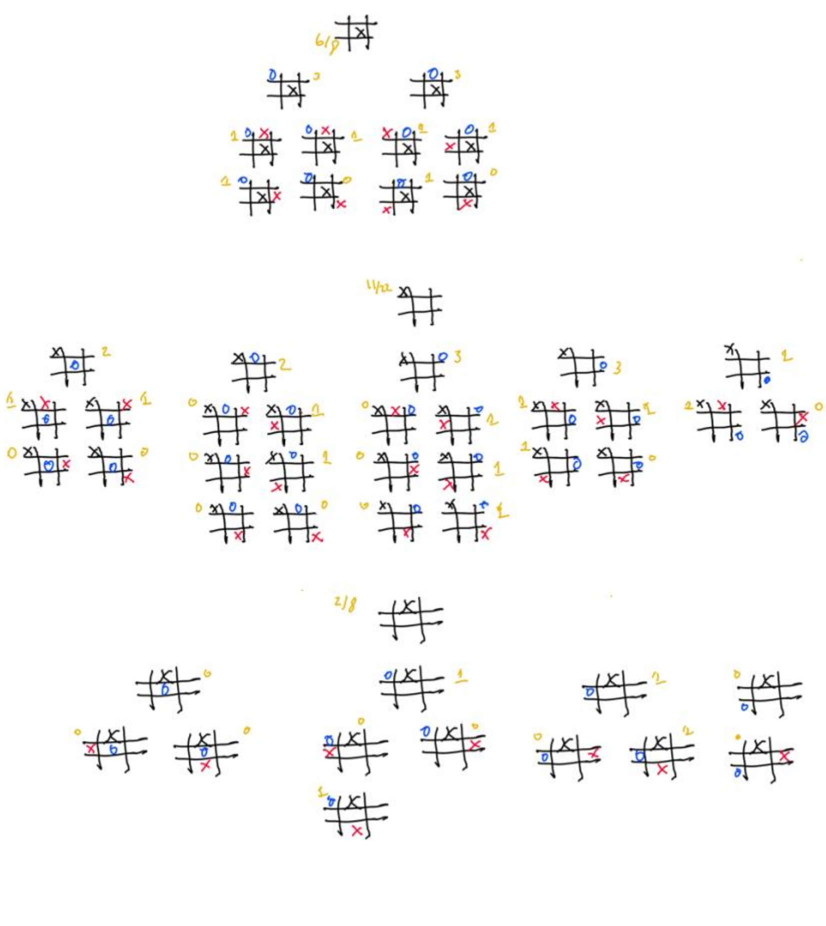
Evan Edelstein

EN.605.645.82.SP26

Module 5 – self check

1) A good heuristic for tic-tac-toe could be to sum all the positions that would win the game for the current player and subtract off the positions that would cause the opponent to win. In this way we reward boards that tend towards wins and penalize boards that tend towards loses. We can calculate the heuristic by iterating over each space on the board and assessing if two pieces are next to it in each cardinal direction and across the diagonals. [note as we will see in (2) this is not a good heuristic since it just becomes 1 or 0 in a 3-play game, including more information in the heretic could better tease out better boards]
2) As a child I always thought that placing the piece in the middle was the best strategy, but as this graph shows the corner has more game states with higher heuristic score.



3) Following the algorithm in the lectures and self-check we can remove strategies by finding dominating strategies. This requires comparing a strategies payoff to all other strategy payoffs. From the perspective of Bar 2, we would look for any column whose payoffs for Bar 2 are greater than all other strategies. We would then do the same for Bar 1, but compare rows. Starting with Bar 2 we can compare strategy 2's payoffs (10, 14, 14) with strategy 4 (12, 20, 28) and see that strategy 4 is always better than strategy 2. This means we can remove the first column from consideration.

		Bar 2					
		\$2		\$4		\$5	
	\$2	10	10	14	12	14	15
Bar 1	\$4	12	14	20	20	28	15
	\$5	15	14	15	28	25	25

We can now switch to Bar 1's perspective and compare strategy 2 (14,14) with strategy 4 (20,28) and strategy 5 (15,25) and see that strategy 2 is dominated by strategy 4, so we can remove the strategy 2 row.

		Bar 2			
		\$4		\$5	
	\$2	14	12	14	15
Bar 1	\$4	20	20	28	15
	\$5	15	28	25	25

Now switching to Bar 2. We can compare strategy 4 (20, 28) and strategy 5 (15, 25). We see that strategy 4 dominates strategy 5, so we can remove the strategy 5 column.

		Bar 2			
		\$4		\$5	
Bar 1	\$4	20	20	28	15
	\$5	15	28	25	25

Switching to Bar 1's perspective, we can compare strategy 4 (20) and strategy 5(15) and remove strategy 5 as is it dominated by strategy 4.

		Bar 2	
		$4	
Bar 1	$4	20	20
	$5	15	28

This results in the result of the SEDS algorithm being strategy 4 for both bars with a payoff of 20 each.

This does resemble state-space search, we could formulate SEDS as a state-space search with the initial state being the full payoff matrix. The states are then different configurations of the matrix, with rows or columns removed. We can then reframe a dominating strategy as a row or column where the values in the row/column are greater than the corresponding value in the other rows/columns. We define an action of removing the dominated row or column from the matrix and generate transition states by returning the states with eliminated rows removed and states with the eliminated columns removed. We could use BFS to find all possible successive eliminations that lead to a Nash equilibrium.