ANALYSIS OF HEURISTIC FUNCTIONS PROVIDED:

Three heuristic evaluation functions were provided and evaluated in tournament game play :

(1) custom_score(game, player) – an aggressive evaluation function that uses the formula "Score = (own valid moves – 2 * opponent valid moves)" to evaluate the score attached to an evaluated valid game move. Here any multiplier greater than 1 (2 was randomly selected > 1) will have the same effect in this evaluation function. The evaluation is more aggressive because at less number of valid moves (distance being the same between own and opponent moves), the score is higher so that the computer player will follow the opponent aggressively in order to limit its moves. This will happen more quickly towards the endgame. For example, if V1 (own moves) =5, V2(opponent moves) = 3, it is obvious from the formula above the score will be higher than if V1 = 10 and V2=8 (same distance of 2 between own and opponent moves), that is any move leading to lesser number of valid moves will be the preferred next move in the ply

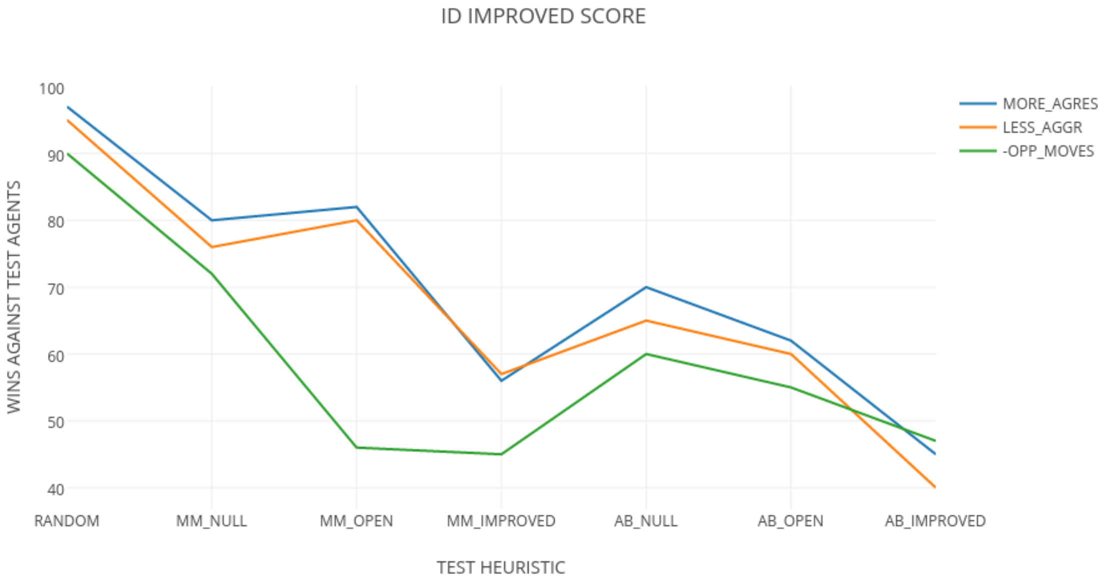
(2) eval_less_aggressive(game, player) – a less aggressive evaluation function that uses the formula "Score = (own valid moves – 0.5 * opponent valid moves) " to evaluate the score attached to an evaluated valid game move. The evaluation is less aggressive in the opening game and also maintains distance between own and opponent moves. This evaluation will explore more positions which could throw up some winning moves by evaluating more sub-trees. Here any multiplier less than 1 (0.5 was randomly chosen) will have the same effect mentioned above.

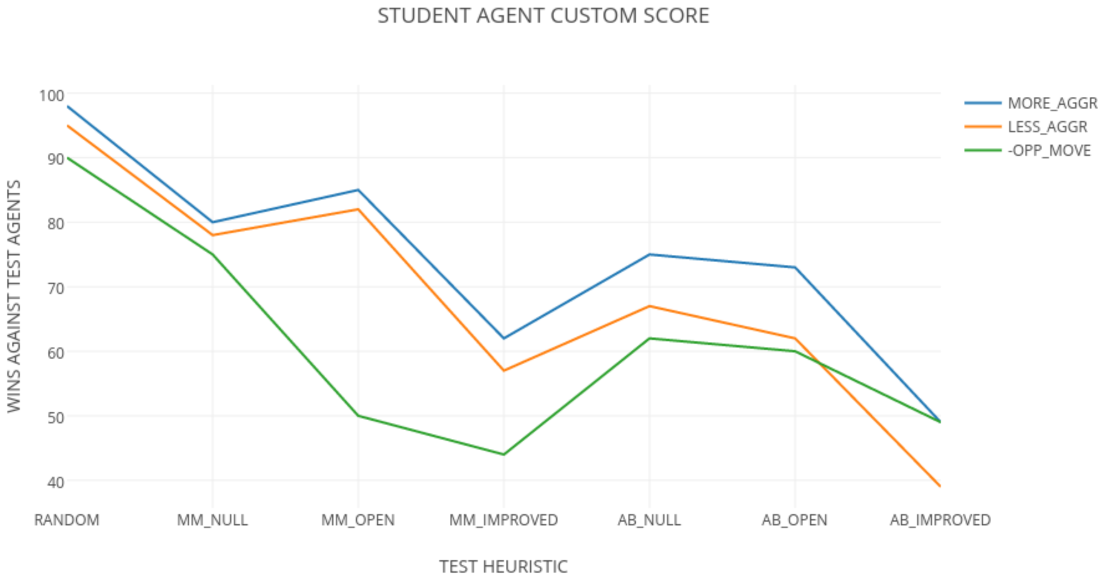
(3) eval_less_oppo_moves(game, player) – an evaluation function that uses the formula "Score = (– opponent valid moves)" to evaluate the score attached to an evaluated valid game move. As seen this evaluation may provide moderately aggressive scoring if the number of valid moves by the opponent are very low especially in the endgame. However it does not use the number of own moves which could be even lower and in which case the student agent could end up losing in some cases when cornered.

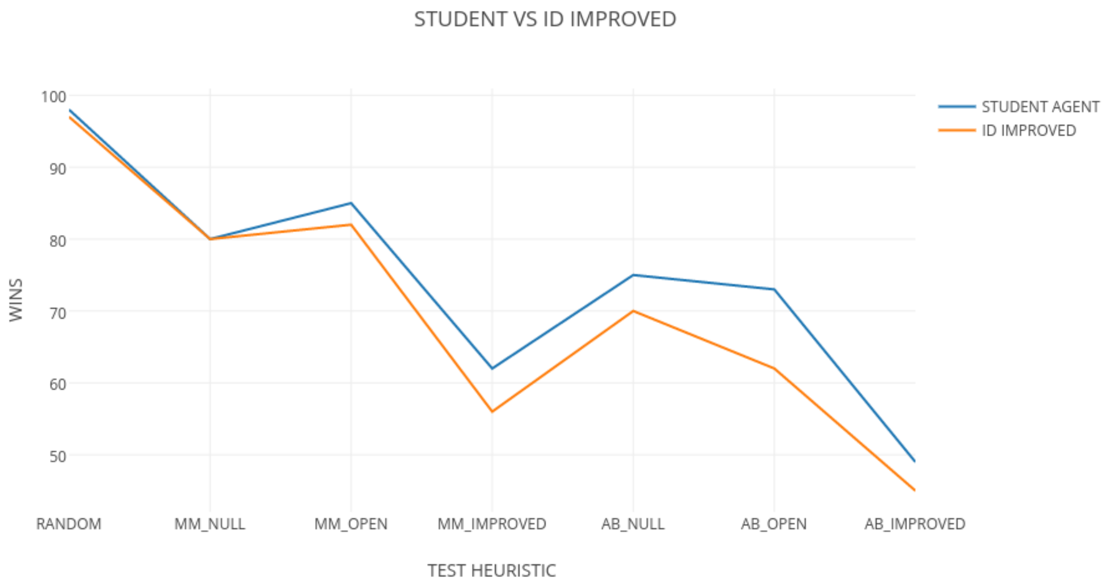
Results of evaluation:

The results are provided below:
The first 2 graphs show the performance of Iterative Deepening with improved score in different runs and Student agent with custom score against the test

heuristics and the third graph shows the performance of Student agent(custom score) vs ID agent(improved score) in 5 matches(data tables are provided below)






STUDENT VS ID IMPROVED
STUDENT AGENT
ID IMPROVED
WINS
100
90
80
70
60
50
RANDOM
MM_NULL
MM_OPEN
MM_IMPROVED
AB_NULL
AB_OPEN
AB_IMPROVED
TEST HEURISTIC

Student/ID agent wins	MAT1	MAT2	MAT3	MAT4	MAT5
RANDOM	**20 / 19**	**19 / 20**	**20 / 19**	**20 / 19**	**19 / 20**
MM_NULL	**18 / 17**	**15 / 17**	**17 / 13**	**15 /16**	**15 / 17**
MM_OPEN	**16 / 17**	**18 / 13**	**16 / 18**	**17 / 17**	**18 / 17**
MM_IMPRO	**13 / 12**	**14 / 11**	**12 / 11**	**11 / 12**	**12 / 10**
AB_NULL	**15 / 12**	**17 / 14**	**14 / 14**	**17 / 14**	**12 / 16**
AB_OPEN	**17 / 12**	**14 / 14**	**15 /11**	**15 /13**	**12 / 12**
AB_IMPROVED	**10 / 9**	**10 / 6**	**9 / 11**	**8 / 10**	**12 / 9**

Table 1: Tournament wins (Student with custom_score (more aggressive eval) and ID agent vs test heuristic)
Each match = 20 games

Student/ID agent wins	MAT1	MAT2	MAT3	MAT4	MAT5
RANDOM	**20 / 19**	**18 / 20**	**20 / 19**	**20 / 19**	**17 / 18**
MM_NULL	**18 / 16**	**14 / 17**	**15 / 13**	**14 /14**	**17 / 16**
MM_OPEN	**16 / 17**	**16 / 13**	**15 / 18**	**17 / 17**	**18 / 15**
MM_IMPRO	**11 / 12**	**12 / 10**	**12 / 11**	**10 / 12**	**12 / 12**
AB_NULL	**12 / 12**	**14 / 10**	**14 / 13**	**12 / 15**	**15 / 15**
AB_OPEN	**11/ 12**	**12 / 14**	**15 /11**	**11 /13**	**13 / 10**
AB_IMPROVED	**9 / 8**	**8 / 6**	**9/ 9**	**7/ 8**	**6 / 9**

Table 2: Tournament wins (Student with eval_less_aggressive score fn and ID agent vs test heuristic)
Each match = 20 games

Student agent/ID agent wins	MAT1	MAT2	MAT3	MAT4	MAT5
RANDOM	**19 / 19**	**19 / 18**	**18 / 17**	**17 / 18**	**17 / 18**
MM_NULL	**18 / 17**	**15 / 12**	**14 / 13**	**15 /13**	**13 / 17**
MM_OPEN	**10 / 8**	**11 / 11**	**8 / 9**	**11 / 8**	**10 / 10**
MM_IMPRO	**10 / 10**	**9 / 11**	**8 / 11**	**9 / 6**	**8 / 8**
AB_NULL	**13 / 12**	**13 / 10**	**12 / 11**	**12 / 14**	**12 / 13**
AB_OPEN	**12 / 10**	**14 / 9**	**12 /11**	**10 /11**	**12 / 12**
AB_IMPROVED	**10 / 9**	**10 / 6**	**9 / 11**	**8 / 10**	**12 / 11**

Table 3: Tournament wins (Student with eval_less_oppo_moves score fn and ID agent vs test heuristic)
Each match = 20 games

RESULT:

It is clear that the custom score employing the more aggressive score evaluation gets more wins over several matches against the test heuristics than the other two evaluation functions and additionally is also better than the ID Improved agent from the lectures. Its win percentage is 74.57% vs 70.28% for the ID improved agent across all matches. The more aggressive evaluation function that causes the computer play to follow the test heuristic agent is therefore preferred to the less and moderatively aggressive agents as well as the improved score agent from the lectures and this is what is therefore selected for the custom_score function eventually.