Go to the Table 1 Group A example technical skills and make sure I use all the ones I possibly can and research them well and incorporate them into the technical solution.

15 marks are for completeness of the code.

and 27 marks are for technical skills in coding!!!!!!!!

SO MAKE SURE YOU FULLY 100% HAVE AS MANY OF THEM AS HUMANLY POSSIBLE!!!!!!

<https://filestore.aqa.org.uk/resources/computing/specifications/AQA-7516-7517-SP-2015.PDF>

Need to start revising for the exams at the start of the year aswell!!!!!!!!!!!

13/08/19

-The Strategy the AI uses is not unbeatable, its pretty good but it will choose an unbeatable strategy if better heuristics are chosen, which I will not do InshAllah yet because the heuristics should be made for the UTTT game, since they are very different to heuristics of the TTT game.

Yes, the reason the AI is not unbeatable is because the heuristics havent fully developed. Once the MCTS reaches a two way win node, the some of the children node of it have a 100% win rate with the current heuristics while the others have a 50% win rate, this is because on the human player's turn in the simulation, the heuristics dont force the simulation to choose the move that will always stop the two way win from occuring.

This would be done by making the heuristics always choose a move that has the least amount of twos for the opponent and the most twos for current player.

-When about to win, AI chooses not to,

problem solved,

there was a bug where a node with a win\_value of ((0.5, 0, or 1) , 1) was not updated itself but adding on its win value to its value, but it was back-propogated, which meant the node had an infinite UCT value while having a 0,0 value, which meant the other child nodes would only get picked once each, and meant that the node with the win value would never be picked for the make\_move function.

Need to address problem 2, and then move on to making the UTTT game, and then updating the code to accompany it properly and fully.

And then I need to update heuristics of it completely, and find online the best heuristics for this, by comparing to other MCTS UTTTs and doing online research aswell as by using my own head.

Then I need to make it work unbeatably, and make difficulty levels for it.

Finally I can decide if I should make sure it works for both players, and allow it to flip a coin to decide who goes first, or I can just make sure the code is completely and utterly perfect for getting a high A or A\* and emailing it once complete to teachers, and then updating it as much as is humanly possible so its completely and utterly perfect and there is no chance it wont get a high A or A\*

All the while doing all these things I need to research career paths and courses, get an idea of what I like doing, research those career paths and talk to Maira and Laiba about that stuff.

Go to the Table 1 Group A example technical skills and make sure I use all the ones I possibly can and research them well and incorporate them into the technical solution.

15 marks are for completeness of the code.

and 27 marks are for technical skills in coding!!!!!!!!

SO MAKE SURE YOU FULLY 100% HAVE AS MANY OF THEM AS HUMANLY POSSIBLE!!!!!!

https://filestore.aqa.org.uk/resources/computing/specifications/AQA-7516-7517-SP-2015.PDF

Need to start revising for the exams at the start of the year aswell!!!!!!!!!!!

1-Next step is to make the game work with the larger UTTT board.

2- Then make the AI work with the larger UTTT board.

3- Look into jobs and courses properly

When human player plays first, AI acts wierd and has problems.

I still need to compare this to the other online MCTS algorithms to see if im doing MCTS right or not.

Dont want to be using the entirely wrong approach and then have to do it all over because it isnt even the MCTS algorithm, and its some mix between MCTS and minimax.

Need to address problem 2, and then move on to making the UTTT game, and then updating the code to accompany it properly and fully.

And then I need to update heuristics of it completely, and find online the best heuristics for this, by comparing to other MCTS UTTTs and doing online research aswell as by using my own head.

Then I need to make it work unbeatably, and make difficulty levels for it.

Finally I can decide if I should make sure it works for both players, and allow it to flip a coin to decide who goes first, or I can just make sure the code is completely and utterly perfect for getting a high A or A\* and emailing it once complete to teachers, and then updating it as much as is humanly possible so its completely and utterly perfect and there is no chance it wont get a high A or A\*

All the while doing all these things I need to research career paths and courses, get an idea of what I like doing, research those career paths and talk to Maira and Laiba about that stuff.

1/9/19

Ive corrected the search function so it stops searching when a node that has any unexpanded child nodes is reached.

I need to next make it so that the UTTT rules are applied using previous moves, which will not slow down anything I do not think, since a prev move variable will be used, and since the ‘is the local grid full will be checked outside of the actual move, something like this.

Once UTTT rules are applied I need to check how well it works, and improve it by improving speed and hence iterations, heuristics, and making sure the whole things works as it should, and gives good results.

I can pretty easily implement difficulty levels after this is completed.

And then I can make sure all good techniques and coding stuff is used in accordance with the spec.

1/9/19

Check how this guy stored the game\_state, thats the key to understanding his code:

https://github.com/chaitanya100100/AI-Agent-for-Ultimate-Tic-Tac-Toe/blob/master/source/agent.py