Algorithm Analysis And Design

Not just a simple ***chess AI*****.....**

TOC

Changes we did after our last meet

One Of the major change was the dropping of the idea of making a mobile application, instead we made a PC java application. Basically we just stopped using Android Studio in the middle of our project which did lead to some complications and obviously made us to feel the shortage of time.

Reason:

The main difficulty was we were at the end of october and first week of november and we still had a lot of work to be done which seemed quite unrealistic for us to do. As all the xml files were still left to be built and some other simple modifications.

Moreover Android Studio wasn't working out properly, it requires a lot of RAM, and since almost all of us are on dual boot systems android studio made our laptops extremely slow. We tried it on an i7 processor.

Things we did and what we said

01 As also suggested by the initial project model, a simple chess AI was made…….

02 The two modes of chess, No as such option but its just a change in the code to be done (changing the mini-max depth)

03 In order to make human feel that they can also beat AI :p, we implemented Quantum chess with piece superposition.

04 A vision for N- Dimensional Chess, sounds a bit crazy like how to visualize it. But don’t worry we are here with a documentation on that(As told in the final proposal)

05 Incorporating a variant of chess like dice chess and moreover proving a documentation on how the Chess can help in visualising some real life problems.

Normal Chess

A very simple chess AI using mini-max algorithm along with the modification of Alpha-Beta pruning to it is designed in just around 800 lines of code elegantly written in java. The evaluation function involves the use of piece square tables.

As all of this sounds pretty trivial, we have made a documentation on whatever changes we could have made and what all are already there.

Quantum Chess

This is pretty much the most important keyword in our chess engine.

A Quantum Chess AI and UI is built considering that whenever a move is made, the chosen piece will have equal probability to be on either place (the initial one and the final one). Each piece is represented in 4 qbits. Each piece has two possible states that are classical and superposed state.

Puzzles(Our work + possible extension)

As of now, we have implemented a feature such that the user can add his own puzzles and we will solve it for him. Moreover there are also around 10 inbuilt puzzles with which one can play around in free time.

A possible extension to this puzzle part was we can design puzzles not only for chess but also for some other paper or board games like tic tac toe, othello. As the codes for these is pretty similar to chess when it comes to UI. Like in tic tac toe, there is just a change in the board size from 8*8 to 3*3

Our Failures during the project.....

The biggest failure that occurred to us was almost in the middle of the total project duration that did have an impact on the overall project. This was our failure with Android Studio which also lead to the change of our original proposition of mobile application to a desktop application :` (

Although the codes we wrote in Android Studio are also present in the final project submission.

Documentations (all our work in 6 documentations)

This was one of the most difficult part of our project as we have to compress all the stuff we read during this two month adventure in just 6 documentation in a easy and understandable format. But we tried our best to ace this part.

- Normal Chess Ai: This one was prepared by Kartik, and is just the things we did for designing the chess AI (Normal one). It involves a little description of the UI the code and other stuff.
- Changes possible: This one was prepared by Aaradhya and involves some of the additions that could have been added to the AI code and also those possible changes that can be made to the existing code.
- N-dimensional Chess: One of the things that make us a bit unique from others. We have made a documentation on N-dimensional Chess about how one can implement it and its visualisation. This was prepared by Priyansh.
- Quantum Chess ppt/Documentation: Designed by Aaditya. It involves almost all of the details related to the Quantum chess part of our project and what all were the references.
- Chess in Automated Driving: Designed by Shreyash and involves how one can visualise real world driving as a chess game. It is a beautifully written document involving the levels of automation possible.
- Chess in TBS games: TBS games is class of games involving all the turn based Strategy games. Chess can very beautifully help in excelling these games. This one was designed by Harshit.

Project Members....

1. Aaditya Sharma (2019113009)
2. Aaradhya gupta (2019114010)
3. Harshit sharma (2019101083)
4. Kartik garg (2019101060)
5. Priyansh GUpta (2019101080)
6. Shreyash rai (2019101096)

So that's it this was our project, thank you.....

Bubie.... ;)