

**WIZARDS’ CHESS**

Team Members : Akshay Kulkarni

Archana Gahiwad

Project Mentors - Kartik Paigwar

Anish Gupta

Aim - Build a GUI based, Voice Controlled Application for Chess.

Software used :

* Eclipse IDE (Java)
* Sphinx4 Library for Java

Parts of the Project -

* Speech Recognition
* GUI development
* Code to control and track chess, and link all 3 parts.

Block Diagram -



Speech Recognition -

* Found a few APIs like IBM Watson, Google Speech to Text API, and CMU Sphinx.
* Issues - IBM’s and Google’s APIs are cloud-based, not available offline. They aren’t open source and charge money as per usage. CMU Sphinx is open source and free, but it’s documentation is difficult to understand.
* Solution - Decided to use MATLAB and carry out speech recognition for digits on our own.
* Learnt audio file handling in MATLAB as a result.
* Learnt noise elimination methods to use before processing speech. Example : LMS Adaptive Filter, Fourier transforms method, etc.
* Issues - Complexity of the filters makes them difficult to understand and implement.
* Solution - Could not find any solution, so skipped to next step of speech recognition.
* The next step is to process the speech signal to extract information from it. There are 5 steps involved in this. For more information, you can refer to the Week 1 presentation.
* Issues - Complexity of the procedure makes it difficult to understand and implement it in MATLAB.
* Solution - Using an open source speech processing toolbox based on MATLAB maintained by Imperial College, London to generate the Mel Frequency Cepstral Coefficients(MFCC) mentioned in the final step.
* The next step is to use the o/p of processing to compare 2 signals. MFCC is a matrix generated after processing. So, we take the (element-wise) difference of the 2 MFCC matrices of the 2 audio signals. The absolute value of the average of all elements of the difference matrix must be very small if audios are similar.
* Issues - Recognition by this method yields mixed results, i.e. some numbers are recognized fairly accurately, while others are not.
* Solution - Could not find any solution. We decided to try CMU’s PocketSphinx API, but it does not take microphone input natively and requires RAW files. So, we moved to Sphinx4 library in Java also maintained by CMU.
* Learnt basic Java, i.e. not including polymorphism, inheritance, etc. Implemented basic speech recognition using this library.
* Issues - Using Eclipse IDE on Ubuntu, code compiles and runs, but does not go beyond a certain point and does not recognize any input.
* Solution - The problem was actually with Ubuntu (and Linux systems in general, but not MacOS) which do not have proper support for microphones. So, switched to Windows and the code ran correctly.
* Issues - Recognition is still not very accurate, but better than in MATLAB.
* Solution - No solution found yet, but probable cause is that language model is built for American accent. Training the model actually requires huge amount of data, which is not possible.
* Next step was to limit the speech recognition output to be valid iff it contains 4 words. This was done to eliminate the generation of random outputs on no input.
* To improve accuracy, changed the dictionary and language model files from default to those based on a corpus file containing 2 digits per line. Accuracy only improved marginally.

GUI Development -

* Installed QT Creator software in Ubuntu. Learnt some basics to design and animate in qml file.
* Designed a chessboard in qml file.
* Learned C++ to write code for giving commands to perform animation in qml file.
* Performed animation in qml file successfully for all pieces.
* Issues - Couldn’t write function to pass parameter for final coordinates of pieces and couldn’t handle qml file by passing commands from C++ file.
* Solution - We shifted from QT Creator to Java for developing GUI of chess.
* Installed eclipse software for Java programming.
* Learned some basics, animation and how to write methods and add classes.
* First wrote method for chessboard designing and adding images.
* Then wrote method for animation of pieces. The coordinates of pieces are stored in 2 arrays x and y. The numbers are gives to each piece and and each box on board. In this method only final and initial coordinates of box are to be given as input.
* Added some extra features to the main frame of game like profiles of players and colour change indication for currently playing player.
* Issue - In animation method for calling repaint method the paint method should be present in the class. So, animation method was not running successfully.
* Solution - Added new class drawpanel in the main frame and then called repaint method in the animation part of code.
* Issue - The main frame of the program is in the animation method hence every time when animation method is called a new frame opens.
* Solution - For solving this problem, a new method needs to be written for animation in which main frame is not necessary. Could not find any such method yet.

Rest of the code -

* First started writing code in C++ as GUI was to be implemented in C++ using QT Creator.
* Wrote validation function to determine validity of input move.
* Then, when GUI was shifted to Java, shifted all C++ code to Java.

Algorithm for any validation function

1. Validation method first determines whether input move is valid on an empty board, i.e. whether the move is legal irrespective of the position of other pieces.
2. Passing this criterion, the path between the initial and final position is checked to ascertain the legality of the move.
3. Passing this criterion, the final position is checked as it should not contain a piece which has same colour, or is the king of opposite colour.
4. Only after passing all these conditions, the move is deemed valid.

* Such methods are written for all pieces, of which the queen method simply calls the bishop or rook method as movement of queen is a combination of the movements of a rook and a bishop.
* Added tracking class with methods to update the 2D array and to store its contents in a text file. It can be used to store multiple instances of the board and implement the Undo function.
* It also contains methods and variables to indicate no. of moves, piece capture, king under check and so on.
* Implemented methods to find whether the king is under check, checkmate, stalemate, draw, etc.
* Added methods to store the board in memory and also to load it back into the 2-D array.
* Finally, added code for special cases of chess like castling, 50 moves rule, etc.
* Added code to main method to control the game, GUI as well as speech recognition.
* Issues : The code gave an exception which did not allow use of speech recognizer more than once in the same program. This is strictly an issue on Windows only, as confirmed by Sphinx4 developers on StackOverflow.
* Solution : First tried modifying the library itself by decompiling and recompiling the class files after editing. But, this method did not work. After reviewing the problem and changing the structure of main program, solved the problem.
* Finally, debugging and testing was required to correct minor errors in the game’s functioning.

References -

* Java

[2-D array in Java](https://stackoverflow.com/questions/12231453/syntax-for-creating-a-two-dimensional-array)

[Global variables in Java](https://stackoverflow.com/questions/4646577/global-variables-in-java)

[File handling in Java](https://www.tutorialspoint.com/java/java_files_io.htm)

[Rename a file in Java](https://stackoverflow.com/questions/14970110/how-to-rename-an-existing-file)

* GUI development

Java

[Java 2D games tutorial](http://zetcode.com/tutorials/javagamestutorial/)

[Java code for chess](https://www.youtube.com/watch?v=h8fSdSUKttk&list=PLOJzCFLZdG4zk5d-1_ah2B4kqZSeIlWtt)

[Drawing chess board](https://www.youtube.com/watch?v=no4m-TIX-rc&t=4s)

For QT creator

[Designing a chess-board](https://www.codeproject.com/Questions/1086527/Design-a-chess-board)

[Already built board](https://www.planet-source-code.com/vb/scripts/ShowCode.asp?txtCodeId=13749&lngWId=3)

[Built board](https://github.com/ssudake21/CHESS-in-Qt)

[Tutorials for QT Creator](https://www.youtube.com/watch?v=aMUh9DmFLto&list=PLD0D54219E5F2544D)

[QT animation tutorials](http://doc.qt.io/qt-5/examples-animation.html)

* Speech Recognition

For MATLAB

[Read and write audio files in MATLAB](http://in.mathworks.com/help/matlab/import_export/read-and-get-information-about-audio-files.html?requestedDomain=in.mathworks.com)

[Record and play audio files in MATLAB](https://in.mathworks.com/help/matlab/import_export/record-and-play-audio.html)

[Comparing 2 audio files](https://stackoverflow.com/questions/17010516/how-to-detect-how-similar-a-speech-recording-is-to-another-speech-recording)

[Removing noise from audio file (LMS filter)](https://in.mathworks.com/matlabcentral/answers/106510-how-do-i-remove-background-noise-from-a-sound-wave)

[Removing noise from WAVE file in MATLAB](https://stackoverflow.com/questions/24195089/remove-noise-from-wav-file-matlab)

[Cross-correlation to help align 2 signals](https://in.mathworks.com/help/signal/ref/xcorr.html)

[Paper on speech recognition of digits using MFCC and DTW.](http://www.irdindia.in/journal_ijaeee/pdf/vol1_iss1/11.pdf)

[Speech Processing Toolbox for MATLAB](http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html)

[Dynamic time warping function in MATLAB](https://in.mathworks.com/help/signal/ref/dtw.html)

For CMU Sphinx

[CMU Sphinx tutorial](https://www.youtube.com/watch?v=WiE_w48umug)

[CMU Sphinx tutorial 2](https://www.youtube.com/watch?v=IAHH6-t9jK0)

[CMU Sphinx Installation](https://www.youtube.com/watch?v=plK6meRVaZ4&t=519s)

[Sphinx 4 and Java usage tutorial](https://procurity.wordpress.com/2016/09/10/make-your-own-voice-command-app-using-java-and-sphinx4/)

[Sphinx4 documentation(not required)](http://my.fit.edu/~vkepuska/ece5526/SPHINX/Sphinx4.pdf)