

**DANIEL ABEL**  
**MET CS521 Information Structures with Python**  
**Professor Eugene Pinsky**  
**Facilitator Anatoliy Aleksandrov**

## Project Summary

For my project, I opted to create a Chess move calculator. In the game of Chess both players have a finite number of moves available to his or her disposal for a given board. It is theorized that in the midgame section of Chess maximizing the number of moves available to you while minimizing the number of moves available to your opponent will help you gain an upper hand before moving to end game. With this in mind, I thought that creating a script that could quickly calculate the number of moves on a Chess board might be useful when analyzing which player has an upper hand in a seemingly even board.

Chess is a complex game with many rules and nuances that must be accounted for. For the scope of the project, I just focused on being able to account for all the moves in a game and sum them together. No further analysis on who had the upper hand in a board was made, the application was just used to sum boards. For reading chess boards from a file, I chose Forsyth-Edwards Notation, since it is easily human readable which allowed for simplest debugging. For more information on the design, board files, and running tests, please see the README.md file.

I did a lot of testing for my project, ranging from Unit to Functional tests. Unit testing was done with the unittest module of python. Unit testing focused mainly on the individual chess piece behavior and board calculation, though other modules were unit tested as well. Integration testing occurred in chessboard.py and boardInitializer.py where I created code to only run if executed as the main method. These tests allowed for quick sanity checks that the board was working as expected. Functional end to end testing of the entire system was done in main.py. In this module I created code that would continuously loop asking for .fen files to evaluate and allowing the user to see the move sets of pieces on the board.

In the end, I would consider the project successful. I can count the number of moves on a board. The calculation is relatively quick, though I did not do performance timing. To continue this project, I would start building a machine learning algorithm that uses number of moves as a possible input.