ZChess Engine

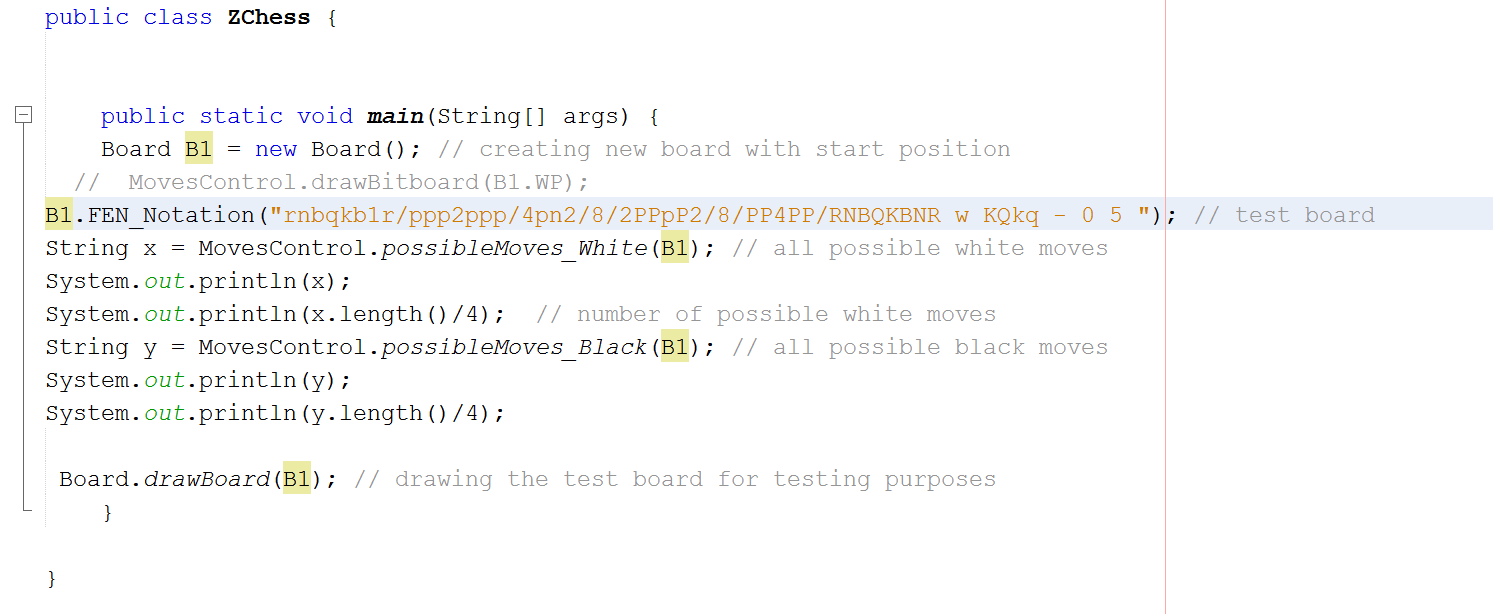
A data class was created (Board) that holds all the information about the board including information about turns and castling. It also implements FEN notation (used in testing different boards)

We used 12 Long variable (Bitboards) to represent a whole board. 6 bitboards for each player (1 bitboard for each piece type)

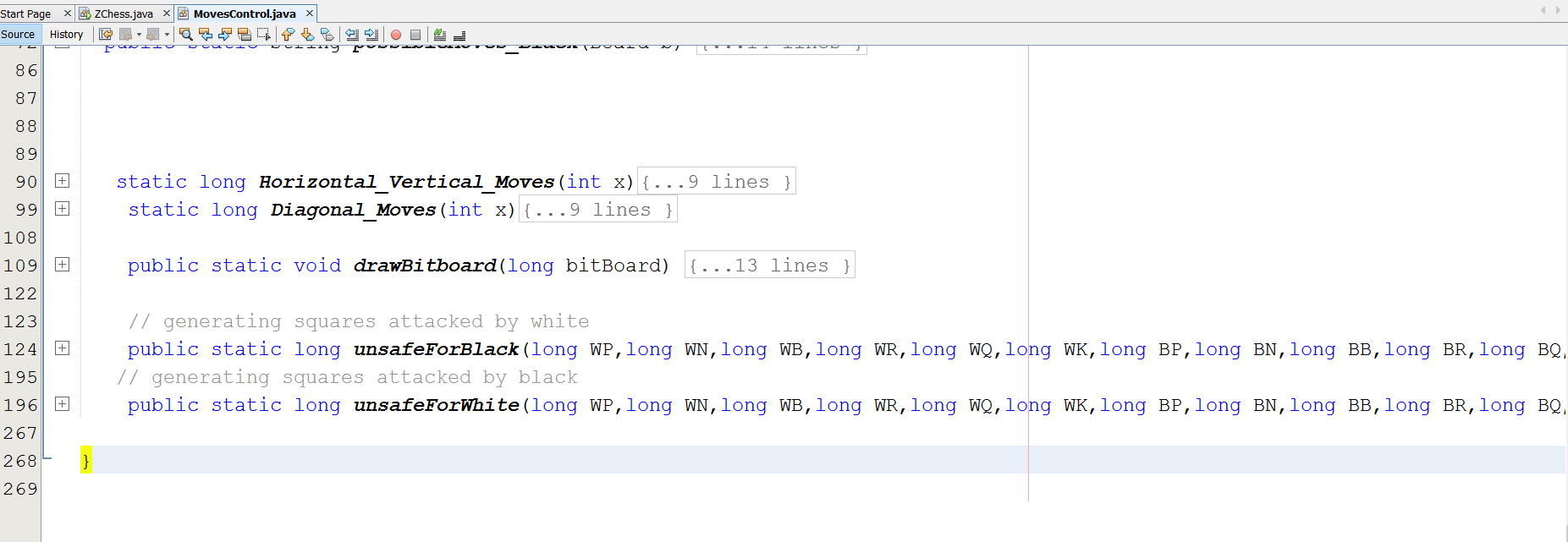
Bitboards are extremely fast and they are much better than other representations like matrices (they are 64 bits and basic operations like shift and logical operations are used to calculate moves later on)



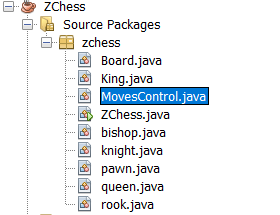
The main class calls a class (MovesControl.java) that gets all the possible moves for white or black (to be used in the evaluation class)



The MovesControl class has some utility methods that help the pieces’ classes to obtain general moves (for example: diagonal and antidiagonal moves)



There is a class for each piece that returns its possible moves



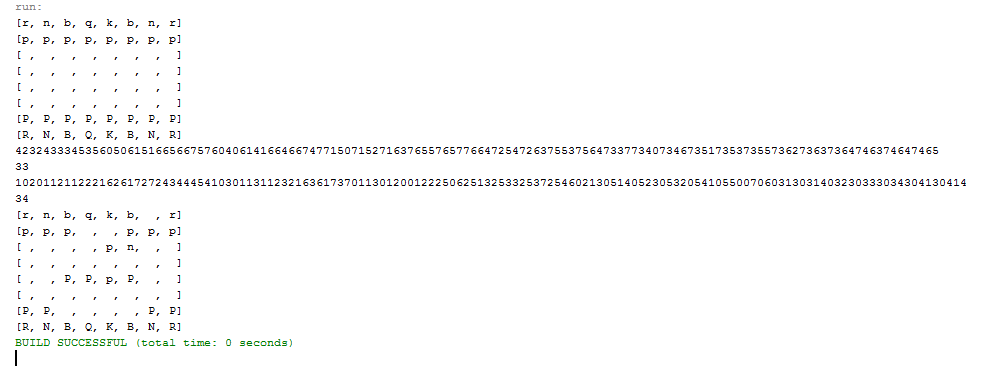
Test: here is a test with a board setup we got from a chess GUI and the program calculates all moves precisely

The moves are on the following notation

Y1X1Y2X2

Y1X1 = current location

Y2X2 = location after move



The number after each moves string is the total number of moves