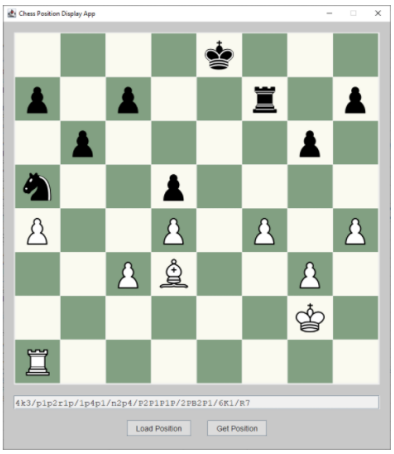
Assignment 8

The deadline for submission of this assignment is **9 pm on Wednesday 14th of April 2021**. This is a strict deadline with automated submission of what work is in the system at that time.

Write a Java GUI application using Swing which allows the user to enter a string that represents the positions of pieces on a chess board and then displays the board position graphically. Your completed application will appear similarly to the screen capture below. Forsyth–Edwards Notation (FEN) is to be used to specify the position in text form.



Capblanca-Tartakover, New York 1924. Move 27. White to move.

Your application will have the following features:

- A chess board that can display pieces. The images for the pieces are supplied to you as a set of PNG image files linked here
- A text field that the user can type (or paste) a FEN position into
- A set position button which, when clicked, displays pieces on the board according to the FEN position in the text field
- The ability to move a piece on the chess board, as follow:
  - When the user clicks a square that contains a piece, the square is highlighted to indicate it has been selected. Only squares that have pieces can be selected/highlighted. If the user clicks a square that has already been selected, then the square is deselected (highlight removed)
  - When a square has been selected and the user clicks another square, the piece on the selected square is moved to the new square. A piece can be moved to any square. If the destination square already has a piece, that piece is replaced (captured)
- A get position button which, when clicked, retrieves the position of the pieces on the board and sets the FEN position text accordingly

See this short video of how the application should function.

Here are some positions in FEN notation you can use to test:

```
R7/6k1/P5p1/5p1p/5P1P/r5P1/5K2/8
b1k4r/p4ppp/4n3/1R6/8/8/PPP2P1P/2KR4
3r4/4k1p1/3pp2p/2p2p2/r1P5/3KPP2/P2R2PP/3R4
8/2R5/p3k1p1/nr4P1/3PKP2/2B5/8/8
```

Structuring your code

Your code should be well structured and clear. The following is a suggested set of classes to implement and an outline of what each should provide. You are not restricted to this structure once your code is well structured and the application functions as specified.

- **ChessApp** – the main application class which sets up the GUI elements
- **Board** – the chess board which inherits from JPanel and contains a 2D array of Square objects. A Board can be loaded with pieces from a FEN notation string. A FEN notation string of the current board can be retrieved and displayed in the text field. A board can have a piece moved from one square to another
- **Square** – a class that inherits from JPanel and implements a mouse listener to detect when a square is clicked. A Square can be highlighted (or have highlighting removed)
- **Piece** – a class that inherits from JLabel and contains an image icon loaded with a PNG file.

Aim to keep your classes under 100 lines of code each. You may add additional classes if needed.

What you will submit

1. Submit your source code as a set of separate .java files (do not submit any other file type – do not submit a zip file)
2. Submit a runnable JAR file of your compiled code. Ensure you follow the instructions here (file updated: 2/4/2021) to create your JAR file so that the grader can run it on their own machine