Software Development Methods Experiment (16302190)

Deadline: 23:59, December 3, 2016 (Sunday)

Project Guideline

The project of the Software Development Methods Experiment course aims at evaluating your software development skills for object-oriented programming in Java. During the development period, the concepts taught in the Software Development course should be taken into account.

Please carefully read the following steps explaining what you need to do to complete your project:

- Build a group of 4 students and inform me on the day of lecture, 15 November 2016 (Tuesday)

- For submission, all the project files should be provided

    - Use **github.com** as your project management/hosting service. I can directly reach all the project files through each group's github project page

- Besides that, a report should be delivered, including:

    - A software design referring to its subsystems, classes and internal connections, with a discussion on why you need those subsystems together with classes and the reasons behind the connections.
        * The programming workload among the group members should be fair
        * Each group member should also contribute to the overall design and the report preparation. The contribution of each member should be explicitly mentioned
    - A set of representative screenshots of the running system, with some test-cases

Online Xiangqi
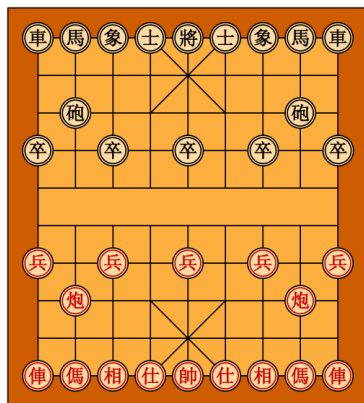
Xiangqi, a.k.a. Chinese chess, is a strategy board game for *two players*. The game represents a battle between two armies, with the object of capturing the enemy's general (king). Distinctive features of xiangqi include the cannon (pao), which must jump to capture; a rule prohibiting the generals from facing each other directly; areas on the board called the river and palace, which restrict the movement of some pieces (but enhance that of others); and placement of the pieces on the intersections of the board lines, rather than within the squares.

Check **https://en.wikipedia.org/wiki/Xiangqi** for the game's rules.

- Should be played online through a graphical user interface

- A bot player should be added so that a single player can play against the computer

- A menu should be present for the features like re/-starting/exiting a game, undoing a move etc.

- Each game should be saved, requiring a user login, so that the played games can be accessed later

- **Bonus**: a chat module for the players to talk