Com S 472/572 Fall 2020

Project 1: A Checkers-Playing Agent (100 pts)

Due at **5:00pm**

Friday, Oct 16

1. Task

English draughts, also called American checkers, are a popular game played by two opponents on opposite sides of an 8X8 gameboard. It is a game on which AI witnessed one of its earliest successes, as best evidenced from a self-learning checkers program written in 1959 by Arthur Samuel, a pioneer in computer gaming who also popularized the term “machine learning”.

In this project, your task is to write a Java program capable of playing checkers against a human player. For rules of the game, please visit Wikipedia (https://en.wikipedia.org/wiki/English_draughts). For a quick tutorial, you may also watch a YouTube video (e.g., https://www.youtube.com/watch?v=ScKIdStgAfU).

To help your implementation and also to make this computer game appealing, we have provided some GUI code within a template such that the human player just needs to move pieces on the GUI, thus saving the effort of typing. A sample display of the board with the first legal move is shown on the next page.

Nevertheless, you do have the freedom of going ahead with an implementation that does not make use of the provided template or GUI. You may also make use of the AlphaBetaSearch implementation in the online code repository of the AIMA textbook.

Below is a list of requirements:

a) Construct a general alpha-beta game playing agent that takes the current state as input and returns a move/action to be made by the agent.

b) Implement an evaluation function that takes a state of the game as input and returns an evaluation value.

c) Implement a move generator function that takes a state as input and returns a list of legal moves at the state.

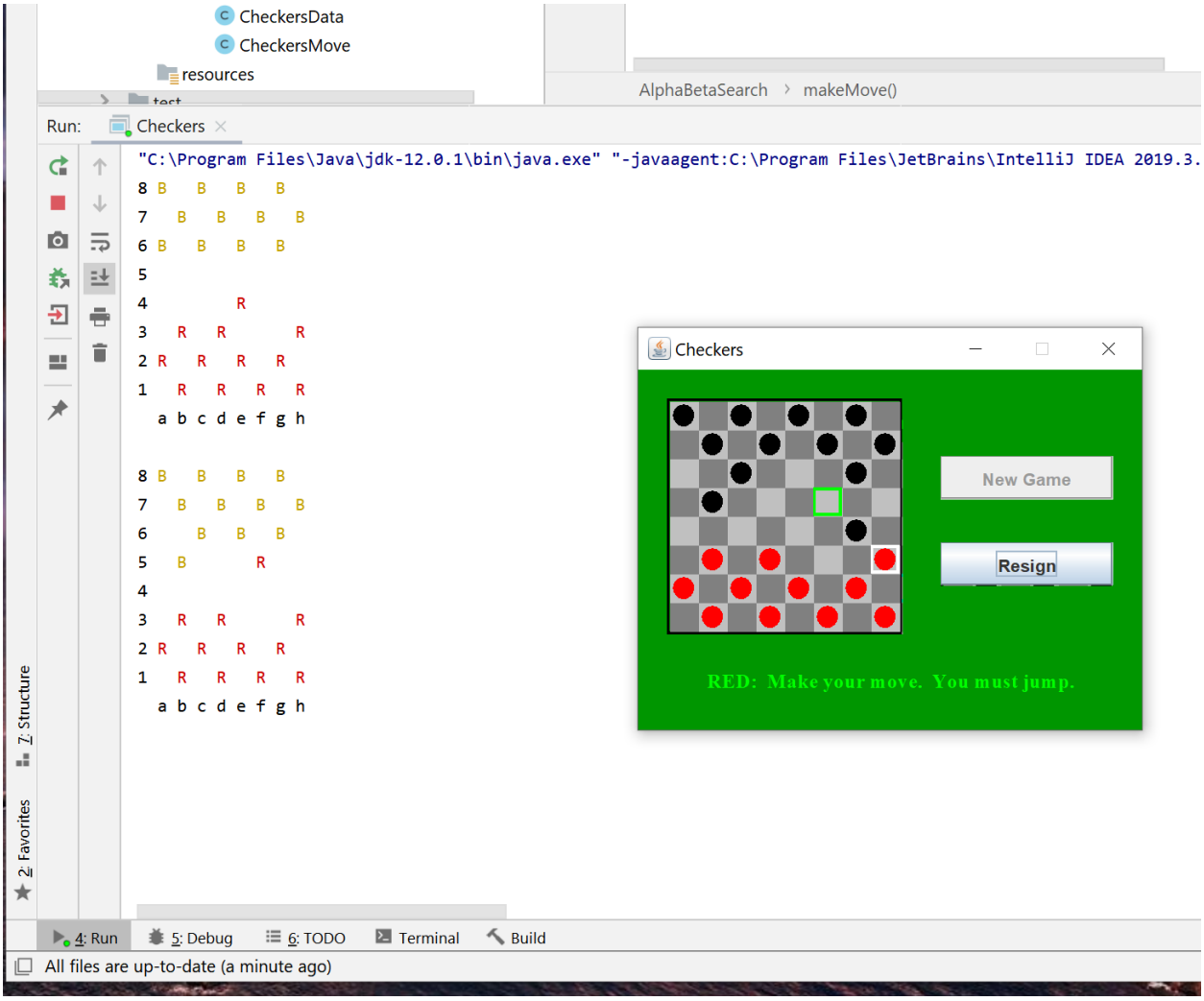
d) If not using GUI, you may print the 8x8 board in the console.

e) Your main function should be able to do the following:

    a. Take as input a move from the user.
    b. Update the board with the user's move.

c. Output the agent's move from the alpha-beta search.
d. Update the board with the agent's move.
e. Repeat the steps until the end of the game.

6. Compare the effect of increasing search depth and improving the evaluation function.



Board display generated by the GUI

2. Submission

Write your classes in the `edu.iastate.cs472.proj1` package. Turn in a zip file that contains the following:

a) Your source code.

b) A short report in PDF on comparisons required in part 6 of the task.
c) A `README` file (optional) for comments on program execution or other things to pay attention to.

Please follow the discussion forums Project 1 Discussion and Project 1 Clarifications on Canvas. Include the Javadoc tag `@author` in each class source file. Your zip file should be named `Firstname_Lastname_proj1.zip`.