CmpE 150 Introduction to Computing
Fall 2017
Project 3—Due: 15/12/2017, 10:00

This is an individual project. Please do not get help from your friends. Do not use Arrays class or Recursion for this project.

Scenario: You will develop a two-player board game. The board contains 16 pieces that are organized in a 4*4 square. Each piece can be either black or white, tall or short, square or round, and hollow or solid. The game starts with an empty board. Players take turns in playing. At each round, a player picks a piece and gives it to the other player, who then places it on the board. A player wins the game after a move if there are four pieces that share one property horizontally, vertically, or diagonally. You can name each piece with the four properties such that "BTSH" means a black, tall, square, and hollow.

Example 1 :

```
E E E E
E E E E
E E E E
E E E E
```

Initially, the board has all the pieces are empty (denoted with "E") as in Example 1). Assume that the program picks "BTSH" and hands it to the user to play, who then places it to coordinates 0,0. Then the board will look like:

Example 2 :

```
BTSH E E E
E E E E
E E E E
E E E E
```

Assume that the user picks "WSSH" and hands it to the program, which then places it to coordinates 1,1. Then the board will move to:

Example 3 :

```
BTSH E E E
E WSSH E E
E E E E
E E E E
```

The game will continue until either all the positions are filled or that there is winning configuration such as the one in Example 4, where the diagonal contains four pieces that are all square.

Example 4 :
BTSH E E E
E WSSH E E
E E BSSH E
E E E WTSS

Example 5 shows another winning configuration, where this time all four horizontal pieces are white.

Example 5 :
BTSH E E E
WTSH WSSH WTSH WSRH
E E BSSH E
E E E WTRS

Tasks: You will write a program that plays this game with a user. Your program needs to play the game, following the rules. If your program wins the game, then you will get a bonus. Implement a program which does the following in that order:

- Ask the user if she wants to start a new game or if she wants to continue an ongoing game. If the answer is a new game, load the board as in Example 1. Otherwise, read the board configuration from an input file (called input.txt). In both cases, display it on the screen. After each move, re-display the current configuration.
- Ask the user to pick a piece using the representation above; e.g., "BTSH". If the piece is already on the board, ask the user to pick another piece. Let the program place the piece. Then, make the program pick a piece and tell the user what it is. Finally, ask the user to place it on the board by entering coordinates. The user will make the moves based on a coordinate system, where 00 is the top left corner and 33 is the right bottom corner. For example, to make the move in Example 2, the player needs to issue "0 0". Make sure there is a space between each.
- When the game ends, print who has won.

Submission: You will submit a project report and your code over Moodle. Your program should be named with your initials and your student number together (e.g., OS2013800027).

Project report should consist of five sections. These are:

1. Problem Description: In this section, you should describe the problem in your words.
2. Problem Solution: In this section, you should specify the concepts (methods, for loop, etc.) that you use in your program. Explain each one (i.e. why you need it, what you accomplish by using it, so on.). Report how many for loops you use.
3. Implementation: This section will include your whole code with comments. You need to pay attention to indentation in order to improve readability.

- Do not forget to explain each variable that you use (i.e. int count=0; // count is the number of items).
- Before each method, specify what the method does (i.e. /* This method . */)

4. Output of the program: A screen-shot of your program output should be put in this section.
5. Conclusion: You should evaluate your work here. State whether you have solved the problem correctly. If not, state what is missing, what could have been improved, and so on.

You will submit these over Moodle as a single zip file where the file name is your student number. Your zip file should consist of your .java file and your report in .doc or .pdf format.

Partial Submission: If you cannot do everything above, you should still submit your code as well as your report. Try to do as much as possible.

Late Submission: Any submission after the deadline is considered late and will not be accepted.