Project 1: Power Connect Four

DUE: Thursday, September 9th at 11:59pm

Extra Credit Available for Early Submissions!

Basic Procedures

You must:

- Fill out a readme.txt file with your information (goes in your user folder, an example readme.txt file is provided)
- Have a style (indentation, good variable names, etc.)
- Comment your code well in JavaDoc style (no need to overdo it, just do it well)
- Have code that compiles with the command: **`javac *.java`** in your user directory without errors or warnings
- Have code that runs with the commands: **`java PowerConnectFourGUI [InputFILE]`**

You may:

- Add additional methods and variables, however these methods **must be private**.

You may NOT:

- Make your program part of a package.
- Add additional public methods or variables
- Use any built in Java Collections Framework classes anywhere in your program (e.g. no ArrayList, LinkedList, HashSet, etc.).
- Use any arrays anywhere in your program (except the data field provided in the **`Column`** / **`PowerConnectFour`**).
    - When you need to expand or shrink the provided data field, it is fine to declare/use a "replacement" array.
- Alter any method signatures defined in this document of the template code. Note: “throws” is part of the method signature in Java, don’t add/remove these.
- Alter provided classes that are complete (**`Token, PowerConnectFourGUI`**).
- Add any additional import statements (or use the “fully qualified name” to get around adding import statements).
- Add any additional libraries/packages which require downloading from the internet.

Setup

- Download the **`p1.zip`** and unzip it. This will create a folder **`section-yourGMUUserName-p1`**;
- Rename the folder replacing **`section`** with the **`001`**, **`002`**, **`003`**, **`005`** based on the lecture section you are in;
- Rename the folder replacing **`yourGMUUserName`** with the first part of your GMU email address;
- After renaming, your folder should be named something like: **`001-krusselc-p1.`**
- Complete the **`readme.txt`** file (an example file is included: **`exampleReadmeFile.txt`**)

Submission Instructions

- Make a backup copy of your user folder!
- Remove all test files, jar files, class files, etc.
- You should just submit your java files and your readme.txt
- Zip your user folder (not just the files) and name the zip **`section-username-p1.zip`** (no other type of archive) following the same rules for **`section`** and **`username`** as described above.
    - The submitted file should look something like this:

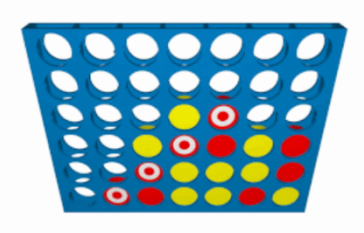
```
001-krusselc-p1.zip --> 001-krusselc-p1 --> JavaFile1.java
                                            JavaFile2.java
                                            JavaFile3.java
                                            ...
```

- Submit to blackboard.

Grading Rubric

Due to the complexity of this assignment, an accompanying grading rubric pdf has been included with this assignment. Please refer to this document for a complete explanation of the grading.

Overview



There are three major components to this project:
1. Implementing one the most fundamental data structures in computer science (the dynamic array list).
2. Using this data structure to implement a larger program.
3. Practicing many fundamental skills learned in prior programming courses including generic classes.

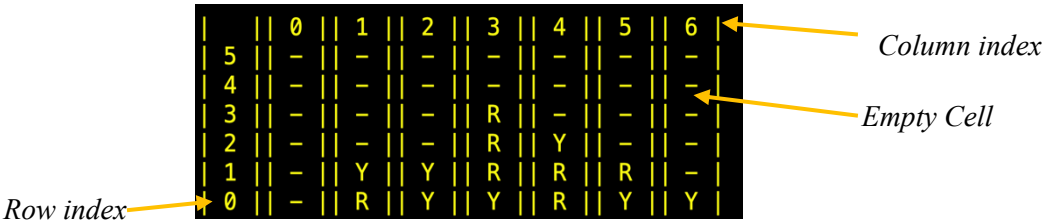
The end product will be a program showing steps to play a Connect Four game (picture shown on the right, Image source: https://en.wikipedia.org/wiki/File:Connect_Four.gif). Connect Four is a game where players win by placing four of their pieces in a row, either horizontally, vertically, or diagonally. The board is a vertical plane with a limited number of columns; playing a piece in a column will drop to the lowest available space. Players take turns dropping pieces into columns, trying to get four in a row before their opponent does. In this project, we will implement a Power Connect Four game and hence follow slightly revised rules as detailed below.

Board. We will follow the tradition and use the board with a fixed seven columns. Each column will be implemented as a dynamic array list with these required features:

- We will only keep players' pieces in a column. An empty cell will never consume any column storage. This implies that if a column currently has no pieces yet, the dynamic array storage should report a size zero.
- **We will allow each column to grow as needed**. In other words, there is no upper-bound on the number of pieces a player can drop into a column.

With these features, it is possible that the seven columns in the board have different sizes. However, we will still show a rectangular game board following these rules:

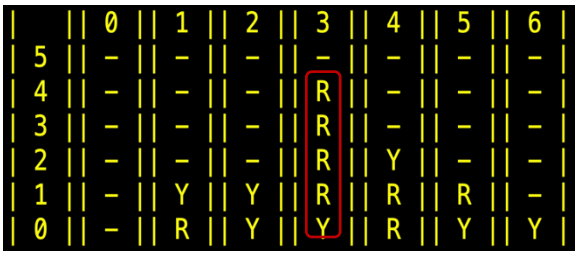
- In display, the game board should always show seven columns and at least six cells of each column.
- If the number of pieces of any column reaches or grows beyond six, we will show all existing pieces plus one more row that is completely empty at the top of the grid. Keeping the top row empty indicates that we allow columns to grow more.
- We number the columns in the ascending order, from left to right (column index). The cells within a column are numbered in the ascending order, from bottom up (row index).



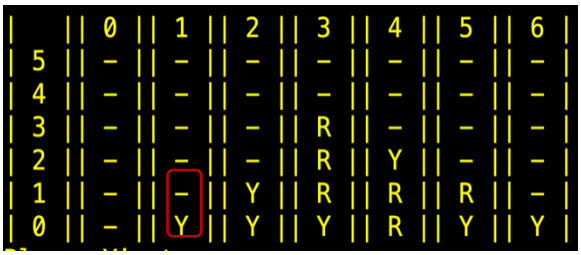
Example 1: A game board with two players (R and Y) and 14 pieces. Column 1 should only store two pieces (R at index 0 and Y at index 1). Column 0 should be completely empty.

Moves. Four types of moves are supported in our Power Connect Four game:

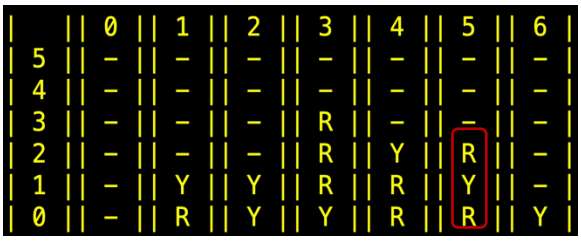
1. **Drop**. The player places a piece into a column; the piece always falls to the lowest open space.
2. **Pop**. If a player has a piece at the bottom (row 0) of a column, the player can remove it. All other pieces in that column would go down by one row as the result. A player cannot pop from a column if the bottom piece does not belong to that player.
3. **PowerDrop.** The player can specify one column and one row to insert a piece. If originally that location had a piece, that piece and all pieces above it must be shifted to make space. We are not allowed to insert a "floating" piece which has a blank spot under it.
4. **PowerPop.** Similar to the pop move, if a player has a piece at a given (column, row), the player can remove it. All pieces above the removed piece would be shifted down to make sure there is no open space between pieces of the same column. A player cannot power pop from a cell if that cell is empty or does not have a piece belonging to that player.



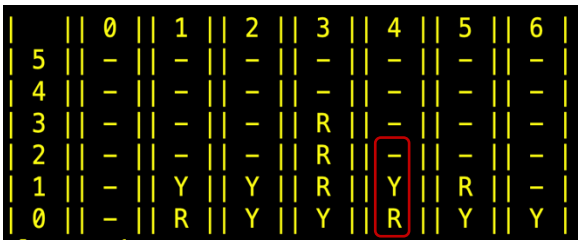
Example 2: Starting from Example 1, R plays drop at column 3; R is the winner



Example 3: Starting from Example 1, R plays pop from column 1



Example 4: Starting from Example 1, R plays power drop for column 5, row 0



Example 5: Starting from Example 1, R plays power pop from column 4, row 1

Players, turns, and winners. We will always have two players with red and yellow pieces respectively. The red player always goes first at the start of the game. The two players must take turns to make valid moves to continue the game. The player who first get four pieces in a row (either horizontally, vertically, or diagonally) is the winner. It is possible that one move will results in both players having a 4-in-a-row at the same time. In that case, the win goes to the player who makes the move. If your move makes the other player having a 4-in-a-row but not yourself, you will unfortunately lose.

Implementation/Classes

This project will be built using a number of classes representing the component pieces of the table we described in the previous section. Here we provide a description of these classes. Template files are provided for each class in the project package and these contain further comments and additional details.

- **Token (Token.java)**: The implementation of game pieces. This class is provided to you and you should NOT change the file.
- **Column (Column.java)**: The implementation of a dynamic array list. We use this as columns of the game board. You will implement this class as a generic class to practice that concept.
- **PowerConnectFour (PowerConnectFour.java)**: The implementation of the Power Connect Four game that can check moves, update board for valid moves, maintain player's turn, and count the connected pieces in a row.
- **PowerConnectFourGUI (PowerConnectFourGUI.java)**: A GUI class to play the game and/or test your implementation. This class is provided to you and you should NOT change the file.

Requirements

An overview of the requirements are listed below, please see the grading rubric for more details.

- **Implementing the classes** - You will need to implement required classes and fill the provided template files.
- **JavaDocs** - You are required to write JavaDoc comments for all the required classes and methods. Check provided classes for example JavaDoc comments.
- **Big-O** - Template files provided to you contains instructions on the REQUIRED Big-O runtime for many methods. Your implementation of those methods should NOT have a higher Big-O.

How To Handle a Multi-Week Project

While this project is given to you to work on over two weeks, you are unlikely to be able to complete this is one weekend. We recommend the following schedule:

- Step 1 (Prepare): First weekend (by 08/29)
  - Complete Project 0 if you haven't.
  - Go over Project 1 with a fine-toothed comb.
  - Get familiar with the game rules.
  - Read about Dynamic Array List (Ch15 of textbook).
  - Think about what to add in the design of the game (**`PowerConnectFour`**).
- Step 2 (**`Column`**): Before the second weekend (08/30-09/03)
  - Implement and test methods in **`Column.`**
  - Complete the design and basic methods of **`PowerConnectFour.`**
- Step 3 (**`PowerConnectFour`**): Second weekend(09/04-09/05)
  - Finish implementing methods of **`PowerConnectFour.`**
- Step 4 (**`Wrapping-up`**): Last week(09/06-09/09)
  - Additional testing, debugging, get additional help.
  - ☺ Also, notice that if you get it done early in the week, you can get extra credit! Check our grading rubric PDF for details.

Testing

The main methods provided in the template files contain useful code to test your project as you work. You can use command like "**`java Column`**" or "**`java PowerConnectFour`**" to run the testing defined in **`main()`**. You could also edit **`main()`** to perform additional testing. JUnit test cases will not be provided for this project, but feel free to create JUnit tests for yourself. A part of your grade *will* be based on automatic grading using test cases made from the specifications provided.

The provided **`PowerConnectFourGUI`** can be run with or without an input file as the command line argument.

- When no file name is provided, the program starts with an empty board and expects input from keyboard. A certain format must be followed to specify the moves. Check sample runs below.
- When a valid file name is provided, the program read the file to get a sequence of moves. It also starts with an empty board. All lines in the file are moves, one per line, following the same format as the keyboard option. Check sample runs below and the provided files for details.
- We provide a number of testing files that you can use with **`PowerConnectFourGUI`** under the folder **input_files**. Check the **README.txt** in that folder for a brief description for each file. Make sure you test more with either keyboard or your own files.
- With keyboard, you can end a game with a special command 'Q'. With or without an input file, the game will terminate either when a winner is identified or the end of the input is reached.

Example Runs (Command Line)

```
>java PowerConnectFourGUI
  Supported Moves:
      D-Drop, P-Pop, PD-Power Drop, PP-Power Pop, Q-Quit
  Example format: 'D 5' - Drop at Column 5
  Example format: 'PP 3 0' - Power Pop from Column 3 Row 0
-----------------------------------------------
- Starting Game
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || - || - || - || - || - || - |
| 0 || - || - || - || - || - || - || - |
Player R's turn
Next Move: D 3
-----------------------------------------------
 1: Move by player R : Drop 3: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || - || - || - || - || - || - |
| 0 || - || - || - || R || - || - || - |
Player Y's turn
Next Move: D 2
-----------------------------------------------
 2: Move by player Y : Drop 2: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || - || - || - || - || - || - |
| 0 || - || - || Y || R || - || - || - |
Player R's turn
Next Move: P 2
-----------------------------------------------
 3: Move by player R : Pop 2: Invalid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || - || - || - || - || - || - |
| 0 || - || - || Y || R || - || - || - |
Player R's turn
Next Move: D 3
```

Sample run with keyboard input.

Moves must follow certain formats. The same formats are used in input files.

Empty grid to start.
R is always the first to make a move.

User typed "D 3" followed by enter.

Valid move: grid updated and switch to the other player Y.

Invalid move: grid not changed; still the same player to make the move.

```
------------------------------------------------
 4: Move by player R : Drop 3: Valid Move
------------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || - || - || R || - || - || - |
| 0 || - || - || Y || R || - || - || - |
Player Y's turn
Next Move: D 2
------------------------------------------------
 5: Move by player Y : Drop 2: Valid Move
------------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || - || Y || R || - || - || - |
| 0 || - || - || Y || R || - || - || - |
Player R's turn
Next Move: D 3
------------------------------------------------
 6: Move by player R : Drop 3: Valid Move
------------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || R || - || - || - |
| 1 || - || - || Y || R || - || - || - |
| 0 || - || - || Y || R || - || - || - |
Player Y's turn
Next Move: D 10
------------------------------------------------
 7: Move by player Y : Drop 10: Invalid Move
------------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || R || - || - || - |
| 1 || - || - || Y || R || - || - || - |
| 0 || - || - || Y || R || - || - || - |
Player Y's turn
Next Move: D 2
------------------------------------------------
 8: Move by player Y : Drop 2: Valid Move
------------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || Y || R || - || - || - |
| 1 || - || - || Y || R || - || - || - |
| 0 || - || - || Y || R || - || - || - |
Player R's turn
Next Move: D 3
```

```
-----------------------------------------------
 9: Move by player R : Drop 3: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || R || - || - || - |
| 2 || - || - || Y || R || - || - || - |
| 1 || - || - || Y || R || - || - || - |
| 0 || - || - || Y || R || - || - || - |
-----------------------------------------------
 Winner: R!
-----------------------------------------------
 - Ending Game
-----------------------------------------------
```

Four tokens in a column: R is the winner.

```
>java PowerConnectFourGUI
  Supported Moves:
      D-Drop, P-Pop, PD-Power Drop, PP-Power Pop, Q-Quit
  Example format: 'D 5' - Drop at Column 5
  Example format: 'PP 3 0' - Power Pop from Column 3 Row 0
-----------------------------------------------
- Starting Game
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || - || - || - || - || - || - |
| 0 || - || - || - || - || - || - || - |
Player R's turn
Next Move: D 1
-----------------------------------------------
 1: Move by player R : Drop 1: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || - || - || - || - || - || - |
| 0 || - || R || - || - || - || - || - |
Player Y's turn
Next Move: D 1
-----------------------------------------------
 2: Move by player Y : Drop 1: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || Y || - || - || - || - || - |
| 0 || - || R || - || - || - || - || - |
Player R's turn
```

Sample run with keyboard input. This sequence has multiple PowerDrop and PowerPop moves.

```
Next Move: PD 1 2
-----------------------------------------------
 3: Move by player R : Power Drop Column 1 Row 2: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || R || - || - || - || - || - |
| 1 || - || Y || - || - || - || - || - |
| 0 || - || R || - || - || - || - || - |
Player Y's turn
Next Move: PD 1 5
-----------------------------------------------
 4: Move by player Y : Power Drop Column 1 Row 5: Invalid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || R || - || - || - || - || - |
| 1 || - || Y || - || - || - || - || - |
| 0 || - || R || - || - || - || - || - |
Player Y's turn
Next Move: PD 2 0
-----------------------------------------------
 5: Move by player Y : Power Drop Column 2 Row 0: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || R || - || - || - || - || - |
| 1 || - || Y || - || - || - || - || - |
| 0 || - || R || Y || - || - || - || - |
Player R's turn
Next Move: PP 1 1
-----------------------------------------------
 6: Move by player R : Power Pop Column 1 Row 1: Invalid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || R || - || - || - || - || - |
| 1 || - || Y || - || - || - || - || - |
| 0 || - || R || Y || - || - || - || - |
Player R's turn
Next Move: PP 1 2
-----------------------------------------------
 7: Move by player R : Power Pop Column 1 Row 2: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || Y || - || - || - || - || - |
| 0 || - || R || Y || - || - || - || - |
Player Y's turn
```

A Power Drop move needs to specify a column index followed by a row index. It works as an insertion.

Invalid: power drop cannot create a floating token.

Invalid: power pop cannot pop from a cell that does not have a matching token.

```
Next Move: PD 1 0
-----------------------------------------------
 8: Move by player Y : Power Drop Column 1 Row 0: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || Y || - || - || - || - || - |
| 1 || - || R || - || - || - || - || - |
| 0 || - || Y || Y || - || - || - || - |
Player R's turn
Next Move: PP 1 1
-----------------------------------------------
 9: Move by player R : Power Pop Column 1 Row 1: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || Y || - || - || - || - || - |
| 0 || - || Y || Y || - || - || - || - |
Player Y's turn
Next Move: Q
-----------------------------------------------
 - Ending Game
-----------------------------------------------
```

You can use move 'Q' to end the game.

```
> java PowerConnectFourGUI input_files/in2.txt
-----------------------------------------------
- Starting Game
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || - || - || - || - || - || - |
| 0 || - || - || - || - || - || - || - |
Player R's turn
Press enter to continue ...
-----------------------------------------------
 1: Move by player R : Drop 1: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || - || - || - || - || - || - |
| 0 || - || R || - || - || - || - || - |
Player Y's turn
Press enter to continue ...
-----------------------------------------------
 2: Move by player Y : Drop 1: Valid Move
-----------------------------------------------
```

Sample run with an input file. Start with an empty grid.

With a file input, it will pause after each move. Press enter to continue.

```
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || Y || - || - || - || - || - |
| 0 || - || R || - || - || - || - || - |
Player R's turn
Press enter to continue ...
------------------------------------------------
 3: Move by player R : Drop 2: Valid Move
------------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || Y || - || - || - || - || - |
| 0 || - || R || R || - || - || - || - |
Player Y's turn
Press enter to continue ...
------------------------------------------------
 4: Move by player Y : Drop 2: Valid Move
------------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || Y || Y || - || - || - || - |
| 0 || - || R || R || - || - || - || - |
Player R's turn
Press enter to continue ...
------------------------------------------------
 5: Move by player R : Drop 3: Valid Move
------------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || Y || Y || - || - || - || - |
| 0 || - || R || R || R || - || - || - |
Player Y's turn
Press enter to continue ...
------------------------------------------------
 6: Move by player Y : Drop 3: Valid Move
------------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || Y || Y || Y || - || - || - |
| 0 || - || R || R || R || - || - || - |
Player R's turn
Press enter to continue ...
------------------------------------------------
 7: Move by player R : Drop 4: Valid Move
------------------------------------------------
```

```
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || Y || Y || Y || - || - || - |
| 0 || - || R || R || R || R || - || - |
-----------------------------------------------
 Winner: R!
-----------------------------------------------
 - Ending Game
-----------------------------------------------
```

Four tokens in a row: R is the winner.

```
>java PowerConnectFourGUI input_files/in6.txt
-----------------------------------------------
- Starting Game
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || - || - || - || - || - || - |
| 0 || - || - || - || - || - || - || - |
Player R's turn
Press enter to continue ...
-----------------------------------------------
 1: Move by player R : Drop 3: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || - || - || - || - || - || - |
| 0 || - || - || - || R || - || - || - |
Player Y's turn
Press enter to continue ...
-----------------------------------------------
 2: Move by player Y : Drop 4: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || - || - || - || - || - || - |
| 0 || - || - || - || R || Y || - || - |
Player R's turn
Press enter to continue ...
-----------------------------------------------
 3: Move by player R : Drop 4: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
```

Sample run with an input file. Start with an empty grid.

```
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || - || - || - || R || - || - |
| 0 || - || - || - || R || Y || - || - |
Player Y's turn
Press enter to continue ...
-----------------------------------------------
 4: Move by player Y : Drop 3: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || - || - || - || - |
| 1 || - || - || - || Y || R || - || - |
| 0 || - || - || - || R || Y || - || - |
Player R's turn
Press enter to continue ...
-----------------------------------------------
 5: Move by player R : Drop 3: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || R || - || - || - |
| 1 || - || - || - || Y || R || - || - |
| 0 || - || - || - || R || Y || - || - |
Player Y's turn
Press enter to continue ...
-----------------------------------------------
 6: Move by player Y : Drop 4: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || - || - || - |
| 2 || - || - || - || R || Y || - || - |
| 1 || - || - || - || Y || R || - || - |
| 0 || - || - || - || R || Y || - || - |
Player R's turn
Press enter to continue ...
-----------------------------------------------
 7: Move by player R : Drop 4: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || - || R || - || - |
| 2 || - || - || - || R || Y || - || - |
| 1 || - || - || - || Y || R || - || - |
| 0 || - || - || - || R || Y || - || - |
Player Y's turn
Press enter to continue ...
-----------------------------------------------
 8: Move by player Y : Drop 3: Valid Move
-----------------------------------------------
```

```
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || - || - || - || - |
| 3 || - || - || - || Y || R || - || - |
| 2 || - || - || - || R || Y || - || - |
| 1 || - || - || - || Y || R || - || - |
| 0 || - || - || - || R || Y || - || - |
Player R's turn
Press enter to continue ...
-----------------------------------------------
 9: Move by player R : Drop 3: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || R || - || - || - |
| 3 || - || - || - || Y || R || - || - |
| 2 || - || - || - || R || Y || - || - |
| 1 || - || - || - || Y || R || - || - |
| 0 || - || - || - || R || Y || - || - |
Player Y's turn
Press enter to continue ...
-----------------------------------------------
 10: Move by player Y : Drop 4: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 5 || - || - || - || - || - || - || - |
| 4 || - || - || - || R || Y || - || - |
| 3 || - || - || - || Y || R || - || - |
| 2 || - || - || - || R || Y || - || - |
| 1 || - || - || - || Y || R || - || - |
| 0 || - || - || - || R || Y || - || - |
Player R's turn
Press enter to continue ...
-----------------------------------------------
 11: Move by player R : Drop 4: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 6 || - || - || - || - || - || - || - |
| 5 || - || - || - || - || R || - || - |
| 4 || - || - || - || R || Y || - || - |
| 3 || - || - || - || Y || R || - || - |
| 2 || - || - || - || R || Y || - || - |
| 1 || - || - || - || Y || R || - || - |
| 0 || - || - || - || R || Y || - || - |
Player Y's turn
Press enter to continue ...
-----------------------------------------------
 12: Move by player Y : Drop 3: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 6 || - || - || - || - || - || - || - |
| 5 || - || - || - || Y || R || - || - |
| 4 || - || - || - || R || Y || - || - |
| 3 || - || - || - || Y || R || - || - |
| 2 || - || - || - || R || Y || - || - |
| 1 || - || - || - || Y || R || - || - |
| 0 || - || - || - || R || Y || - || - |
Player R's turn
Press enter to continue ...
```

Column grows to have 6 tokens: display one additional empty row at the top.

```
-----------------------------------------------
 13: Move by player R : Drop 3: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 7 || - || - || - || - || - || - || - |
| 6 || - || - || - || R || - || - || - |
| 5 || - || - || - || Y || R || - || - |
| 4 || - || - || - || R || Y || - || - |
| 3 || - || - || - || Y || R || - || - |
| 2 || - || - || - || R || Y || - || - |
| 1 || - || - || - || Y || R || - || - |
| 0 || - || - || - || R || Y || - || - |
Player Y's turn
Press enter to continue ...
-----------------------------------------------
 14: Move by player Y : Drop 4: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 7 || - || - || - || - || - || - || - |
| 6 || - || - || - || R || Y || - || - |
| 5 || - || - || - || Y || R || - || - |
| 4 || - || - || - || R || Y || - || - |
| 3 || - || - || - || Y || R || - || - |
| 2 || - || - || - || R || Y || - || - |
| 1 || - || - || - || Y || R || - || - |
| 0 || - || - || - || R || Y || - || - |
Player R's turn
Press enter to continue ...
-----------------------------------------------
 15: Move by player R : Drop 4: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 8 || - || - || - || - || - || - || - |
| 7 || - || - || - || - || R || - || - |
| 6 || - || - || - || R || Y || - || - |
| 5 || - || - || - || Y || R || - || - |
| 4 || - || - || - || R || Y || - || - |
| 3 || - || - || - || Y || R || - || - |
| 2 || - || - || - || R || Y || - || - |
| 1 || - || - || - || Y || R || - || - |
| 0 || - || - || - || R || Y || - || - |
Player Y's turn
Press enter to continue ...
-----------------------------------------------
 16: Move by player Y : Drop 3: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 8 || - || - || - || - || - || - || - |
| 7 || - || - || - || Y || R || - || - |
| 6 || - || - || - || R || Y || - || - |
| 5 || - || - || - || Y || R || - || - |
| 4 || - || - || - || R || Y || - || - |
| 3 || - || - || - || Y || R || - || - |
| 2 || - || - || - || R || Y || - || - |
| 1 || - || - || - || Y || R || - || - |
| 0 || - || - || - || R || Y || - || - |
Player R's turn
Press enter to continue ...
```

Column keeps growing in height.

```
-----------------------------------------------
 17: Move by player R : Pop 3: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 8 || - || - || - || - || - || - || - |
| 7 || - || - || - || - || R || - || - |
| 6 || - || - || - || Y || Y || - || - |
| 5 || - || - || - || R || R || - || - |
| 4 || - || - || - || Y || Y || - || - |
| 3 || - || - || - || R || R || - || - |
| 2 || - || - || - || Y || Y || - || - |
| 1 || - || - || - || R || R || - || - |
| 0 || - || - || - || Y || Y || - || - |
Player Y's turn
Press enter to continue ...
-----------------------------------------------
 18: Move by player Y : Pop 4: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 7 || - || - || - || - || - || - || - |
| 6 || - || - || - || Y || R || - || - |
| 5 || - || - || - || R || Y || - || - |
| 4 || - || - || - || Y || R || - || - |
| 3 || - || - || - || R || Y || - || - |
| 2 || - || - || - || Y || R || - || - |
| 1 || - || - || - || R || Y || - || - |
| 0 || - || - || - || Y || R || - || - |
Player R's turn
Press enter to continue ...
-----------------------------------------------
 19: Move by player R : Pop 4: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 7 || - || - || - || - || - || - || - |
| 6 || - || - || - || Y || - || - || - |
| 5 || - || - || - || R || R || - || - |
| 4 || - || - || - || Y || Y || - || - |
| 3 || - || - || - || R || R || - || - |
| 2 || - || - || - || Y || Y || - || - |
| 1 || - || - || - || R || R || - || - |
| 0 || - || - || - || Y || Y || - || - |
Player Y's turn
Press enter to continue ...
-----------------------------------------------
 20: Move by player Y : Pop 3: Valid Move
-----------------------------------------------
|   || 0 || 1 || 2 || 3 || 4 || 5 || 6 |
| 6 || - || - || - || - || - || - || - |
| 5 || - || - || - || Y || R || - || - |
| 4 || - || - || - || R || Y || - || - |
| 3 || - || - || - || Y || R || - || - |
| 2 || - || - || - || R || Y || - || - |
| 1 || - || - || - || Y || R || - || - |
| 0 || - || - || - || R || Y || - || - |
Player R's turn
Press enter to continue ...
-----------------------------------------------
 - Ending Game
-----------------------------------------------
```

Token popped: column/grid shrink.

No winner but the end of the file is reached.