Brian Wiley

May 8, 2017

CIS163AA

Java Final Project

Title: Basketball Partition

My idea for my final project is to take a list of basketball players on a single team, it can be for any level of expertise from teams that are 8 and under, middle school through college, and even pros. What the program is trying to solve is first off who will be the starting players based on their stat ratings and what position these players will start at considering their individual stats in points per game, rebounds per game, assists per game, steals per game, and blocks per game. This is just the first problem that is solved by the program. The second and more difficult problem to solve is splitting the team into the two most equal teams with the smallest difference between total stats ratings so that they can practice against each other to not only make themselves better but their team better.

I will first briefly discuss the modules from the class room that will be utilized to complete this program. The first module used will be the functionality for scanners. The program initially asked for input directly in the program until I modified it to take input from a TextFile, CSV, or GUI. There are three versions of the final program depending on how the end user wants to enter the data. They all use the same three classes and output the same text files which will be discussed later. The second module will use if/else statements with equality operations to compare many of times the sums of the two team’s total stats ratings to continue pairing options until smallest difference is found. The third module used are for loops and while loops and while loops that contain for loops. This is done much often during the program to more player’s total stats ratings from team to team in various ways and methods to cover the most options. The fourth module used are Arrays and ArrayLists. These are consistently iterated through and repositioned to compare team totals. The fifth module used are methods and classes and uses import/export as a sixth module. Below is the breakout of the classes used and methods. Please note there are three ways to run the program listed in bold. The documents that are underlined are used for all three ways to run the program

**To Run with straight TextFile, use the following class files:**

1. SplitTeamsArrayList.java class file – This runs the program and pulls the text file TotalStatsFile.txt.
2. methodsClass.java class file – This class contains all the methods that do not specifically pertain to the swapping of players from ArrayList to ArrayList to find the smallest difference.
3. mainPartitionMethod.java class file – This class contains only one method. It breaks the teams into two teams, highest stats first and lowest stats second and follows one method until it gets to a difference between teams that is the smallest possible for that method.
4. insideMethods.java class file – This class contains 8 different inside methods that takes the smallest difference from the mainPartionMethod class and completes iterations in each method until the smallest difference is discovered. If the teams are not exactly equal with zero difference it runs all 8 methods.
5. TotalStatsFile.txt (currently filled with 14 players but you can add more)

Files to be produced from running TextFile class file are:

1. OriginalPlayers&Stats.txt – This will show total stat info in order that was entered for confirming the next text files indeed work and with the correct players.
2. SmallestDiffSoFar.txt – This file is continuously updated when the comparison operation for smallest difference of two equal teams is updated and finishes after entire program including all classes and methods are ran.
3. StartingTeam.txt – THIS FILE IS WHAT THE USER WILL USE AS A FINAL SOLUTION RESULT. It contains the starting team of 5 players with the 5 highest total stats ratings and from there takes output from the program for the player with the highest points as shooting guard, highest rebounds as power forward, highest assists as point guard, highest steals and small forward/big guard, and highest blocks as the center.
4. PracticeTeams.txt – THIS FILE IS WHAT THE USER WILL USE AS A FINAL SOLUTION RESULT. It contains the programs output of the two most evenly distributed teams. Here is does not list what positions they should play but I can if you wanted to modify the program to just run what it does for the StartingTeam functionality. I chose not to include this in the first version of the program because, personally I tend to believe you can play different positons in practice and move players around.

**To Run with Excel CSV File, replace the following files:**

1. SplitTeamsArrayListExcel.java class file – Instead uses an Excel CSV BufferedReader to import from CSV file.
2. TotalStatsTest.csv – This file contains the same data as the used for the text file program but is easier for user to understand. They don’t need to enter number for total players at the top and there are nice column headers to indicate what information is needed.

\* Same three methods classes, methodsClass.java, mainPartitionMethod.java, & insideMethods.java.

Files to be produced from running Excel CSV class file are:

\*Same four text files will be produced, OriginalPlayers&Stats.txt, SmallestDiffSoFar.txt, StartingTeam.txt, & PracticeTeams.txt

**To run with GUI Interface, replace the following files:**

1. PlayerInterfaceLoopOutputArea.java class file – This file allows user to enter data within a GridBagLayout to calculate and an Output area will contain the two practice teams. Program can be modified to include the starting team as well. The user can update this GUI and recalculate if they want or made mistakes. Currently only allows entry for 10 players but can be modified easily in the code.

\* Same three methods classes, methodsClass.java, mainPartitionMethod.java, & insideMethods.java.

Files to be produced from running Excel CSV class file are:

\*Same four text files will be produced, OriginalPlayers&Stats.txt, SmallestDiffSoFar.txt, StartingTeam.txt, & PracticeTeams.txt

As indicated by being able to run the GUI this is the seventh module used from the class. Lastly two more modules that are included from class are Collections and Exception handling is required as well for the import/export functionality.

The instructions to run the program are as follows. To run the program of each type you will just need to make sure you have all the following files in one folder:

1. methodsClass.java
2. mainPartitionMethod.java
3. insideMethods.java
4. SplitTeamsArrayList.java and TotalStatsFile.txt to run the text file program. Just open and hit run.
5. SplitTeamsArrayListExcel.java and TotalStatsTest.csv to run the Excel csv program. Just open and hit run.
6. PlayerInterfaceLoopOutputArea.java to run the GUI program. Just open and hit run. Here you need to enter number of players, names, individual stats, and hit the “Calculate” button. You do not need to enter 10 players. You can enter less but to do this you MUST do the following; make sure you enter the correct number of players at the top and when entering you cannot skip an entry. For instance, you cannot enter 6 for number of players but enter information in spaces 1-5 skip 6 and enter the 6th player information in any of the 7-10 spots. If you change from 10 players after you hit calculate to less or from 6 players to more you always need to update the number of players before hitting “Calculate” and if you made a mistake in the individual stats you must again hit “Calculate” to re-calculate.
7. You do not need these four text files that are outputted automatically. They are created for you but if you want you can also have in the same folder: OriginalPlayers&Stats.txt, SmallestDiffSoFar.txt, StartingTeam.txt, and PracticeTeams.txt

There you go. That is basically it. The program can be modified for any sport as well. It currently outputs stats ratings with double value to one decimal point but the number of decimal places can be easily modified. I have also posted this paper and all the files to GitHub at the following address <https://github.com/BJWiley233/Java>.

I can be emailed with any questions at [BRI2237169@maricopda.edu](mailto:BRI2237169@maricopda.edu) or [BJWiley23@hotmail.com](mailto:BJWiley23@hotmail.com).

