**COP 4331- Fall 2013**

**Homework 2**

**Due: October 21, 2013**

* **No late submissions will be accepted**
* **Please submit the homework through Webcourses using the following format: COP4331C\_Fall13\_HW#\_LastName\_FirstName.pdf, where # will be 1, 2, or 3.**
* **This homework assignment is an INDIVIDUAL assignment, not a team effort!!!**

**Problem statement:**

You are required to improve on the system developed in Homework-1. You have been requested to improve on the existing game store and add a suggested game listing. Currently, users have to manually search for games within the game store without knowing what is new, highly rated, and relevant to their current game library. In order to improve the system you have been asked to force the following requirements when storing game information: <name>, <release date>, <genre>, <rating>, and <# of players>. Using this information an automated sorter will generate a list of the top 3 recommended games displayed in the store for an individual to purchase in different categories.

The categories are as follows:

1. New Releases (Sorted by release date with newest games being first)
2. Genre (Displaying the 3 newest games in a given genre)
3. Rating (Displaying the 3 highest rated games, does not take into account release date)

The above 3 categories should also be able to be searched with a multiplayer flag (for example top 3 new multiplayer games)

**Assignment:**

1. You will be provided a .txt file that **you must use** for your submission. This .txt file will essentially be a “database” of games. It will be named “input.txt”. Your program must automatically open and run “input.txt”.
2. Your program should rank games based on the above categories (6 total).
3. Keep in mind that while your program should work for the sample input; it should **also work** for different input that follows the same format (ie: if a game’s rating drops to 0, it should not be the highest rated game).

**Deliverables:**

* All submissions must be a zip file that contains the source code, .exe files and README file including information necessary to run your programs (Hint: think of the readme file as a preliminary attempt at creating a user manual).
* Your program should compile and run. If it doesn't compile and/or crash, you will receive only a partial credit for the code. **IT MUST RUN USING THE GIVEN “INPUT.TXT” FILE.**
* You have two options for language, C++ and Java. If you choose C++, you have two options, you can either submit a makefile or you can submit it as a Visual Studios Project. If you choose Java, you **must** submit the Jar, as well as the source code, or submit a Netbeans Java project. If there are issues with these things, you can let us know and we’ll try to work with you, but only good reasons for exceptions will be considered.

**Constraints of input.txt:**

The first line in input.txt will be a number from 1 – 100 representing the number of games you will be processing

Following that, will be n number of games in the following format:

Game name

Date in the format: xx/xx/xxxx

Genre: One or more of the following genre [RPG / Indie / Action / FPS / Casual]. Genre will be separated by forward slashes ‘/’

Score: (Real number between 1 and 10)

Number of players: (whole number between 1 and 16)

Example:

1

Hello Kitty Island Adventure

10/10/2010

RPG / Indie

9.9

2

**Output:**

Handling output is entirely up to you. You will be providing a “User Manual” in the form of a README.txt file so you are not constrained to an output method.

Examples for ways to handle output could be:

1. User Driven, prompting the user to select which top 3 games they want to see
2. Automatic, running your program either dumps to the screen or to a separate text file the results of analyzing the input file.