Clayton Szelestey

5/8/17

CMPT 220

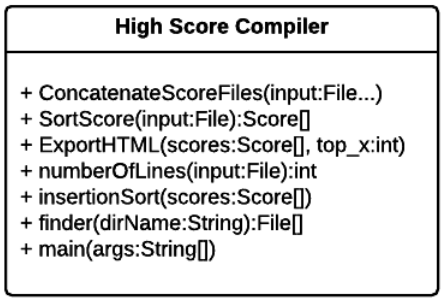
Juan Arias

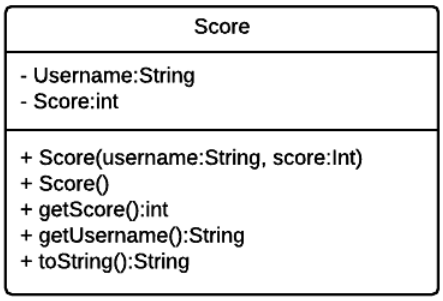
Project 2: Semester Project

Abstract: The program I am creating takes separate files containing high scores and concatenates them into a single high score file. It also produces an html file containing a specified number of top scores.

Introduction: The motivation for this project comes from a separate project I am working on for my Game Design II class. My game has a scoring system and players can earn high scores. I would like to use this program to take high scores from instances of my game on multiple machines and create a global high score list. The outputting in html would be, ideally, so I can easily update a website with the global high scores, and can use this program with any game that has a scoring system as long as the file format is correct. Players would then be able to go to a website and check the global high scores to see where they stand. In this paper, I will go over how I went about implementing this idea.

Detailed System Description:





The main method will execute the program. ConcatenateScoreFiles will find all of the .txt files in the specified directory (in this case it is just the default path, but that is easily adjustable) that are not named “compiledScores” or “Sorted_compiledScores” by means of the finder method, and concatenate them together into a single file that has the same formatting as the original files. Then the SortScore method will take this concatenated file, sort it using the insertionSort method and numberOfLines method, and produce a new file based on the high scores, from highest to lowest. It will also return an array of sorted Score objects. The program then asks how many scores the user would like displayed on their generated leaderboard website. The exportHTML method then takes in the sorted Score array and number of desired high scores to be shown and will produce a webpage. Measures have been taken to ensure that the user is aware if there is an issue with the program. Errors for erroneous score file input and erroneous input for the number of displayed high scores can occur. The format for the score files is a .txt file containing any number of lines with the format “username - 123”, where username is replaced with the player’s username and 123 is replaced with the corresponding high score.

Requirements:

- Take in any number of high score files within a directory
- Concatenate scores from all input into a single file

- Sort high scores from concatenated file, from highest to lowest with help using Score class
- Produce a sorted file of all high scores
- Have new sorted score file be in format that can be used again by the program
- Produce webpage containing specified number of high scores for a leaderboard

User Manual: Put all individual high score files into a specified directory (currently set to java's default). Run the program and answer the prompt asking what number of high scores you would like displayed in the html file. A file containing all global high scores will be output along with the html file containing the specified number of scores to be displayed (e.g. Top 10).

Conclusion: The inspiration for this project came from another project I am working on for my Game Design II class. The game I am working on has a high score system. I would like to be able to take all the high score files saved on individual machines and compile them into a single source for all high scores. The goals of the project are to be able to keep track of high scores across multiple instances of my game and have a viewable leaderboard in the form of a website. The program also produces files that have a format that can be used again by the program, should you want to create a high score list based off several different high score lists.

References:

https://colorlib.com/wp/css3-table-templates/ - HTML Template