

EECS 2510 – Project 3

Due: April 5, 2018

Description:

In this project you will implement a binary tree to hold information on movies. Each movie will hold a title, a year released, an optional MPAA rating, a length (in minutes), and an optional URL. They will be organized into a binary search tree. The key fields for the movie are the title and the year. You will want to write a comparison operators for the movie class that will allow basic Boolean comparisons ($=$, $<$, $>$). You will need to be able to read the initial tree in from a file, add movies, remove movies, list the movies, and write the movies out to a file.

Details:

We will work on a basic outline for the classes involved in class and you may use what we develop in class as a starting point. On startup your application should try to open a file “movies.txt” to input an existing group of movies. The file should be line oriented and consist of:

- Movie Title
- Movie Year
- Movie MPAA Rating
- Movie Length
- Movie URL

The end of the data will be signaled by the end of file. As the movies are read in you should be able to insert them into your binary search tree.

L – The list command, should list all of the movies in the tree using an in-order traversal. That means they will be in alphabetical order.

I – The insert command, should prompt the user for the 5 needed fields for a movie, dynamically create a movie and insert it into your binary search tree.

S – The search command, should take a Title for a movie and a year and, if the movie is in your binary search tree, it should print the movie information. If it is not in your binary search tree it should print an error message reporting the movie could not be found.

D – The delete command, should take a Title and a year for a movie and, if the movie is in your binary search tree, remove the node containing the movie from your binary search tree.

Q – The quit command, should write all the movies in your binary search tree out to “movies.txt” and exit the program.

Hints:

What kind of traversal would be appropriate for outputting the files at the end of the program?

Start with the insert routine and make sure you can insert into a tree. The list command makes sense as the second routine. I would probably put the delete command last in terms of adding it to my code.

Submission:

Your program source code and movies.txt file should be submitted in the spot provided on BlackBoard.