CSCI 2270 – Data Structures - Section 100

Instructor: Shayon Gupta

Assignment 6, Oct 2018

Binary Search Trees II: Electric Boogaloo

OBJECTIVES

1. **Delete nodes in a BST**
2. **Traverse a BST**

Background

This assignment builds off of the previous one. Make a copy of assignment 5, and use it as a starting point. Please use the ***MovieTree.hpp*** file uploaded on moodle for assignment 6, **not** the MovieTree.hpp of the previous assignment 5.

Assignment

- Implement the following functions in addition to the ones from the last lab:
  - ***`void deleteMovie(string title)`***
    This function should find the specified movie in the tree, then delete it. If the movie isn't found, print the following instead:

    ```
    cout << "Movie not found." << endl;
    ```

    Hint: You don't need a recursive helper function, but it will be easiest to use one.

  - ***`void countMovies()`***
    This function should recursively count the total number of movies in the tree, and return the result. The code to print this is below:

    ```
    cout << "Count = " << count << endl;
    ```

    Hint: You will need a recursive helper function for this.

  - ***`void rentMovie(string title)`***
    This function should have the same behavior as it did in assignment 5, with one change - when the last copy of a movie is rented (changing its quantity to 0) it should call the new *deleteMovie* function to delete that movie from the tree.

  - ***`~MovieTree`***
    Remember to delete all the nodes from the tree in the deconstructor. We didn't check for this during the last assignment, but this one will make sure you free up all the memory that is allocated by the tree.

- Like before, your main function should first read information about each movie from a file and store that information in a MovieTree. **The name of the file with this information should be passed in as a command-line argument.** Feel free to keep using *HW5-Movies.txt* on Moodle as a test file. It is in the format:

```
1,<Movie 1 title>,<Movie 1 year>,<Movie 1 quantity>
2,<Movie 2 title>,<Movie 2 year>,<Movie 2 quantity>
Etc…
```

- **After** reading in the information on each movie from the file, display a menu to the user. The code to print this menu is the same as before, but with two additional options:

```
// Display menu
cout << "======Main Menu======" << endl;
cout << "1. Find a movie" << endl;
cout << "2. Rent a movie" << endl;
cout << "3. Print the inventory" << endl;
cout << "4. Delete a movie" << endl;
cout << "5. Count movies" << endl;
cout << "6. Quit" << endl;
```

- Any options that existed in the previous assignment should have the same behavior with the exception of a small change to how movies are rented, which is described above. The new options should have the following behavior:
  - **Delete a movie**
    Call your tree's *rentMovie* function on a movie specified by the user. Prompt the user for a movie title using the following code:

    ```
    cout << "Enter title:" << endl;
    ```

  - **Count movies**
    Call your tree's *countMovies* function.

Submission

Submit your code on Moodle by following the directions on Assignment 6 Submit. You must submit an answer to be eligible for interview grading!