Project: Book Pile

Pre-requisites: Chapter 4

Imagine a pile of books with a special restriction: no book in the pile can be *below* a book with a shorter title (measured by number of characters including spaces):



For this project you’ll create the class BookPile which stores and manages this type of pile as a linked chain. Your program should implement at a minimum the requirements below (though you should add more functions if needed) and meet style guide requirements.

## Starter Files

Included in the supplied starter files are:

* Book.h/Book.cpp – The Book class is nearly identical to the Node class we’ve seen used for linked chains but modified to contain a book title as a string rather than generic template data.
* main.cpp – A main() which does testing of your BookPile class. I recommend you first create stub versions of all the required functions listed below which will allow you to compile.
* Sample Ouput.txt – A sample of the full output of the completed project with the provided main().

You must create the class BookPile as BookPile.h and BookPile.cpp. Note this is not a templatized class.

## BookPile General Requirements:

* Store the pile inside BookPile as a linked chain. The nodes in the linked chain will be Book objects. (The Book class is provided in the starter files).
* Books should be maintained in the linked chain ordered by title length. So, the book with the longest title should always be in the head node, the book with the shortest title in the tail node. You don’t need to use a sorting algorithm to accomplish this… just insert books into the linked chain in the proper place as they are added.
* Books with the same title length may be stacked in any order. *Dune* can be on top of *Rust*, or *Rust* can be on top of *Dune*. Either is legal.

## BookPile Functions:

| BookPile()  Include an appropriate default constructor. |
| --- |
| BookPile(const BookPile&)  Include a copy constructor which does a proper deep copy. |
| int getTitleLength(string)  Private method that returns the number of characters in the parameter including spaces and punctuation. Other methods that need the length of a title should use this helper. |
| bool removeBook(string)  Removes the book with title matching the parameter. Returns true if book was removed, false if it was not.  Some hints:   * You’ll probably need 2 different branches: one for when the book to be removed is the head node in the chain, one for anywhere else in the chain. * When the book to be removed isn’t the head node, you’ll need a pointer to the node *in front of* the node you want to remove to be able to successfully “link around” the node you’re removing. * Draw a picture to help you visualize! |
| bool removeBook(int)  Remove book at the parameter position where 1 is the top of the pile. Returns true if book was removed, false if it was not.  See hints from removeBook(string) as this will be similar. |
| bool addBook(string)  Add book with given title to the pile in the proper ordered location. Should not add the book to the pile if:   * + The title is empty string—books must have titles   + The title already exists in the pile--no duplicate books allowed.   Returns true if book was added, false if it was not.  Hint: As with removeBook(), you’ll need separate branches for adding a new head node and adding a node in any other position in the linked chain. |
| int findBook(string)  Returns the position of the book if it exists in the pile, -1 if it does not. Position numbers begin with 1. |
| void displayPile()  Displays the books in the pile to the console numbered and in order from the top of the pile to the bottom followed by the number of characters in their title.  So, a pile containing the books in the image at the top of this document would be displayed as:   1. Brave New World (15) 2. Dragonflight (12) 3. Neuromancer (11) 4. Contact (7) 5. Cinder (6) 6. Dune (4)   If there are no books in the pile display Book pile is empty. |
| string getBook(int)  Returns the book at the given position number where 1 is the top of the pile. Returns empty string if the position is invalid. |
| bool renameBook(oldName, newName)  Allows the changing of the name of a book (for instance, if there is a typo in the title). Note this may mean the book changes position in the pile. The pile doesn’t change if newName is empty string or if the book with oldName is not found in the pile.  Returns true if the book title was found and replaced, false otherwise. |
| int bookCount()  Returns the number of books in the pile. |
| clear()  Removes all books from the pile. |

## Turn In

Upload your BookPile.h and BookPile.cpp prior to the due date.

Jacob Kier

Johnson County Community College