CS214: Data Structures
Assignment-2: Linked List

Instructions

1- Students will form teams of **2** students (**Can be from different groups).**
2- Deadline of submission is **Monday, December 26th at 11:55 pm.**
3- Submission will be on google classroom.
4- No late submission is allowed.
5- No submission through e-mails.
6- **Please follow the Submission Notes found below**
7- **In case of Cheating you will get a negative grade whether you give the code to someone, take the code from someone/internet, or even send it to someone for any reason.**
8- You have to write clean code and follow a good coding style including choosing meaningful variable names.

Task: Binary Search Tree "FCAI-Reads Application"

Implement a binary search tree which allows the user to add data of books, remove books using their title, display the books sorted by their title ascending or descending, and search for a book by title, by category, or by author.
Note that the BST uses the book title as the key for organizing the tree.

Assuming you have a class Book

```
class Book
{
    string title;
    string author;
    string category;
    int numOfPages;

    public:
      Book(string, string, string, int);
      bool operator<(const Book&);
      void print();
};
```

- Implement the class constructor, print member function, less than operator to compare books using their titles, and setters/getters functions

CS214: Data Structures
Assignment-2: Linked List

- Save the class declaration and implementation in a single header file named as Book.h
- Declare and implement BSTNode and BST in a file named as BST.h.
  Your BSTNode consists of (1) a book object , (2) a left pointer, and (3) a right pointer.
- Include Book.h in BST.h
- Your main will include BST.h

Implement a global **readBooks(istream&,BST&)** function, which takes the file name and the BST and reads the books in the file into the BST. Consider having the following file as a user reading list to use it for filling your BST.

- After inserting the whole 14 books using **readBooks**, show the following options in a menu to the user to choose the desired operation. Each operation calls a member of the class BST.

  1-Display all books sorted by title ascendingly (use **traverseAsc**).
  2-Display all books sorted by title descendingly (use **traverseDesc**).
  3-Remove a book by title (use **remove**).
  4-Add a new book (to be added to the already existing books) (use **insert**)
  5-Search for a book by title (use **searchTitle**)
  6-Display certain author's books (use **searchAuth**)
  7-Display certain category's books (use **searchCat**)

```
14
The 7 Habits of Highly Effective People: Powerful Lessons in Personal Change
Stephen R. Covey
Self Help
372
First Things First
Stephen R. Covey
Self Help
384
The 8th Habit: From Effectiveness to Greatness
Stephen R. Covey
Self Help
432
The 3rd Alternative: Solving Life's Most Difficult Problems
Stephen R. Covey
Self Help
456
Becoming
Michelle Obama
Biography
426
Michelle Obama: A Life
Michelle Obama
Biography
432
The Light We Carry: Overcoming in Uncertain Times
Michelle Obama
Biography
336
The Alchemist
Paulo Coelho
Novels
175
The Zahir
Paulo Coelho
Novels
336
Brida
Paulo Coelho
Novels
224
The Kite Runner
Khaled Hosseini
Novels
371
A Thousand Splendid Suns
Khaled Hosseini
Novels
372
The Monk Who Sold His Ferrari
Robin Sharma
Novels
208
The 5 AM Club
Robin Sharma
Self Help
336
```

CS214: Data Structures
Assignment-2: Linked List

8-Exit

In main you should:

- Read from books.txt to fill BST (insert 14 times)
- Show the menu repeatedly until the user chooses to exit.

Grading Info:

FCAI-Reads Application	**Book Class**	**5**
	BSTNode Class	**5**
	BST Class Declaration	**10**
	readBooks	**10**
	Add book (main and insert)	**10**
	Remove book (main and remove)	**10**
	Display ascending by title (main and traverseAsc)	**10**
	Display descending by title (main and traverseDesc)	**10**
	Remove a book (main and remove)	**10**
	Search by title (main and searchTitle)	**10**
	Search by author (main and searchAuth)	**10**
	Search by category (main and searchCat)	**10**
	Separate Files	**10**
	Total	**120**

Submission Notes:

You will have one folder named by your IDs “ID1_ID2” and it includes 3 code files: book.h, BST.h and main.cpp. Then zip this file and the zipped file name will automatically be ID1_ID2.zip (example: **20202020_21212121.zip**).
Only one of the team members should submit on google classroom.