CS244 Data Structure
Programming Assignment 1
(100 points)
No late submission will be allowed!

Objective:

- Practice on using linked list.

Problem Statement

Write a program that maintains a list of books in stock for a bookstore. The program should *repeatedly* print a menu of options to allow a user to select from the following options, until the user decides to exit.

- Insert a book record (Option1)
- Delete a book record (Option2)
- Print the book list (Option3)
- Search the book list (Option4)

Each book record in the list consists of **book title** (string), **number of copies** in stock (int), and **price** (double). The book list should be maintained in the lexicographical order of the book titles.

Option 1: should read from user a book record (title, quantity, and price) and add the record into the list. (Update quantity and price only, if the book exists)
Option 2: asks for a book title and deletes the record that matches the title.
Option 3: lists all the books stored in the array in the lexicographical order of the book titles.
Option 4: allows a user to type a book title, then find search in the book list. If the matched book title has been found, displays the book record, otherwise, reports the book is not in the stock.

To make it simple, for this program, assume the book title is case sensitive. (e.g., “computer networks” and “Computer networks” are considered as two different books)

Lexicographical ordering of Strings: when comparing two strings, you compare the first characters of each string, then second characters, and so on, until:

↓— One of the strings ends
↓— You find the first character pair that does not match.

If one of the strings ends, the longer string is considered the “larger” one in Lexicographical ordering.

Complete the class definition of the following class BookList and use it in your program. Note you can add other member functions to the class Book and/or the class BookList, if you need them for your program, but you should not remove either data members or member functions from the following declarations.

```
class Book
{
   public:
     string title;
     int quantity;
     double price;
     Book* next;
};

class BookList
{
   public:
     // Constructor
     BookList();
     // Destructor
     ~BookList();

     // Search for the book with the given title t
     // If the book is found, print its title,
     // quantity in stock and price; otherwise report
     // the book is not in the stock.
     void search(string t);

     // Insert a new book record into the list.
     // The book has title t, number of copies in stock
     // is q, and price is p.
     // Note that the vector of books should be maintained
     // in the lexicographical order of the book titles.
     // If the book title already exists in the list,
     // then upgrade the record with q and p.
     void insert(string t, int q, double p);

     // Delete a book record, whose book title is t,
     // If the book is not in the list, do nothing.
     void remove(string t);

     // Print the book records in the vector in the
     // lexicographical order of book titles
     void print();

   private:
     Book* first;             // pointer to first node
     Book* last;              // pointer to last node
     int count;       // total number of books in stock

};
```

The following are some examples when you run the program:

```
This application stores a list of books in stock for UW Stout bookstore.
Select from:
1. Insert a book record
2. Delete a book record
3. Print the book list
4. Search the book list
5. Quit
1
Please enter the title of the book: computer algorithm
Quantity adding to stock: 35
Price: 102.5
Select from:
1. Insert a book record
2. Delete a book record
3. Print the book list
4. Search the book list
5. Quit
3
--------------------------------------------Total: 35 Books in Stock--------------------------------------------
title: computer algorithm               # of copies: 35 price: 102.5
--------------------------------------------------------------------------------------------------------------------------Select        from:
1. Insert a book record
2. Delete a book record
3. Print the book list
4. Search the book list
5. Quit
1
Please enter the title of the book: Computer networks
Quantity adding to stock: 18
Price: 95.8
Select from:
1. Insert a book record
2. Delete a book record
3. Print the book list
4. Search the book list
5. Quit
3
--------------------------------------------Total: 53 Books in Stock--------------------------------------------
title: Computer networks                # of copies: 18 price: 95.8
title: computer algorithm               # of copies: 35 price: 102.5
-----------------------------------------------------------------------------------------------------------------------------
```

```
Select from:
1. Insert a book record
2. Delete a book record
3. Print the book list
4. Search the book list
5. Quit
1
 Please enter the title of the book: computing concepts
 Quantity adding to stock: 125
 Price: 63.0
 Select from:
 1. Insert a book record
 2. Delete a book record
 3. Print the book list
 4. Search the book list
 5. Quit
 3
 ---------------------------Total: 178 Books in Stock---------------------------
 title: Computer networks          # of copies: 18 price: 95.8
 title: computer algorithm         # of copies: 35 price: 102.5
 title: computing concepts         # of copies: 125 price: 63
 --------------------------------------------------------------------------------
 Select from:
 1. Insert a book record
 2. Delete a book record
 3. Print the book list
 4. Search the book list
 5. Quit
 4
 What is the title of the book you want to search for? computer graphics
 "computer graphics" is NOT in stock
 Select from:
 1. Insert a book record
 2. Delete a book record
 3. Print the book list
 4. Search the book list
 5. Quit
 4
 What is the title of the book you want to search for? Computer networks
 "Computer networks" is in stock:
 title: Computer networks
 # of copies: 18
 price: 95.8
 Select from:
 1. Insert a book record
 2. Delete a book record
 3. Print the book list
 4. Search the book list
 5. Quit
```

```
2
Enter the title of the book you want to delete: computer security
"computer security" is not in stock. Deletion failed.
Select from:
1. Insert a book record
2. Delete a book record
3. Print the book list
4. Search the book list
5. Quit
2
Enter the title of the book you want to delete: Computer networks
"Computer networks" has been deleted Select from:
1. Insert a book record
2. Delete a book record
3. Print the book list
4. Search the book list
5. Quit 3
---------------------------Total: 160 Books in Stock---------------------------
title: computer algorithm          # of copies: 35 price: 102.5
title: computing concepts          # of copies: 125          price: 63
-----------------------------------------------------------------------Select       from:
1. Insert a book record
2. Delete a book record
3. Print the book list
4. Search the book list
5. Quit
1
Please enter the title of the book: computing concepts
Quantity adding to stock: 100
Price: 63.0
The book already exists in stock; its record has been upgraded.
Select from:
1. Insert a book record
2. Delete a book record
3. Print the book list
4. Search the book list
5. Quit
3
---------------------------Total: 250 Books in Stock---------------------------
title: computer algorithm          # of copies: 35 price: 102.5
title: computing concepts          # of copies: 225 price: 63
-----------------------------------------------------------------------

Select from:
1. Insert a book record
2. Delete a book record
3. Print the book list
```

```
4. Search the book list
5. Quit
5
   Thank you for using my program. Press ENTER to continue.
```

Submission Guidelines:

To receive full credit, you must follow these guidelines

1. All code must include comments, be properly indented and use descriptive variable names where appropriate.
2. Compress the entire project folder with solution directory to a compressed file and *submit this to the appropriate dropdox on D2L*
3. After submission, double check to make SURE your submit the proper files, ***YOU WILL NOT HAVE A CHANGE TO FIX THIS AFTER THE DUE DATE!!***
   Please ask for help if you need it.

Grading:

1. If your program does not compile and lack efforts (i.e., lack comments and function implements), you receive 0.
2. If your program does not compile but shows significant efforts in code, you receive at most 30% of the total credit.
3. If you program can compile but missing functionalities, you receive about 60% of the total credit. You may receive more than 60% according to the following rubrics.

	Points	Deducted Points
Correctness	60	
Provide the correct header file(s)	5	
Provide the correct implementation of member functions	30	
Program repeatedly display the main menu for user to choose	5	
Program displays the correct output when new book record is inserted, a book record has been deleted, or a book record has been modified, and so on	10	
Program maintains the book list in lexicographical order of the book title, and the total count of book in stock.	10	
Style	20	
Lay out your program in a readable fashion and user-friendliness	5	
Include comments as specified in the lecture notes	10	
Provide separate .cpp files and header files for class(es)	5	
Testing	20	
Testing cases of each option in the memu	10	
Testing of boundary cases. (for example, when book list is empty, when the book record is inserted as the first or last one in the book list)	5	
Testing invalid user input	5	
Your Score		