

CSCI 275 Assignment Week #10

Due: Saturday 4/9/22

Late: Saturday 4/16/22 (5 point penalty)

Points: 60

Task:

This program will practice writing and using a base class and a derived class

Input:

Text file **books.txt** contains a list of books. For each book lists the book type (general Book or Textbook), title, author, and ISBN number.

Textbooks also list the course for which the text will be used. All ISBN numbers are listed with 10 characters.

```
books.txt - Notepad
File Edit Format View Help
Textbook
Programming with C++
Y. Daniel Liang
0136097200
COS221
Textbook
Introduction to Programming
David Schneider
013212856X
COS120
Book
Full Stack Applications
Stephen Prata
1571691316
```

A valid ISBN is determined by the following algorithm. Given the ten digits $d_1d_2d_3d_4d_5d_6d_7d_8d_9d_{10}$ in the ISBN number the last digit d_{10} is called the checksum. If the checksum is 10, character 'X' is used.

d_{10} should equal $(d_1 + d_2 * 2 + d_3 * 3 + d_4 * 4 + d_5 * 5 + d_6 * 6 + d_7 * 7 + d_8 * 8 + d_9 * 9) \% 11$

For example: 0136097200 is a valid ISBN.

$(0*1 + 1*2 + 3*3 + 6*4 + 0*5 + 9*6 + 7*7 + 2*8 + 0*9) \% 11$
 $= (0 + 2 + 9 + 24 + 0 + 54 + 49 + 16 + 0) \% 11$
 $= 154 \% 11 = 0$ which matches the last digit

Hint: You can extract each character, convert to a string with function **to_string**, and use function **stoi** to convert to an integer. You could also do character arithmetic knowing the ASCII code for '0' is 48.

1. The following base class **Book.h** models a book and is available on Brightspace. Do not alter this class specification, your goal is to implement it in **Book.cpp**.

```
class Book
{
public:
    Book(); // POST: empty book
    Book (string t, string a, string i); // POST: book with title t, author a, ISBN i
    // If i is invalid ISBN assign "missing"
    // accessors
    string getTitle( ) const; // POST: return title
    string getAuthor( ) const; // POST: return author
    string getISBN( ) const; // POST: return ISBN

    // modifiers
    void setTitle (string title); // POST: set title to t
    void setAuthor (string author); // POST: set author to a
    void setISBN (string isbn); // POST: set ISBN to i
    // If i is invalid ISBN assign "missing"
    void display (ostream& out) const; // POST: display title, author, ISBN one per line on out
private:
    string title; // book title
    string author; // book author
    string ISBN; // book ISBN
    bool validISBN (string isbn); // POST: return true if ISBN is valid, else false
};
```

2. Write class **TextBook** is derived from **Book** using public inheritance. This class adds a string data member to indicate the course related to the book (ex. COS220). It should contain a default constructor and a parameterized constructor where the user passes in title, author, ISBN, and course. Include accessor and modifier functions for the new data member. Write a **display** function that overrides the base class **display** function to also list the course data member.

3. The application first asks the user of their major (ex. COS). Courses for that major will start with these letters (ex. COS220). Open the **books.txt** file and read the books one at a time, storing the data in the correct type of object (**Book** or **TextBook**). Display information about the book including text "<-- For your major" where appropriate. After all items are read, report the number of books that match the user's major.

Note: There is no need to use an array or other high-level collection this week. You only need one while loop that processes the books from the file. Next week we are going to discuss the issue of arrays/collections as it pertains to an inheritance hierarchy.

Sample Output:

```
Enter your major: BUS
COS221      Programming with C++      Y. Daniel Liang      0136097200
COS120      Introduction to Programming David Schneider      013212856X
Tradebook   Full Stack Applications      Stephen Prata      1571691316
BUS103      Exploring Microsoft Access      Robert Grauer      0135098599    <-- For your major
COS115      Blended HTML, XHTML, CSS      Julie Bojak      missing
Tradebook   Project Management      Mike Dawson      1423902270
COS220      Absolute C++      Walter Savitch      0201709279
Tradebook   Computer Hardware Essentials      Donna Gosselin      missing
BUS200      Database SQL Queries      Aimee Heldon      0618016902    <-- For your major
BUS240      Professional Excel      Rob Bovey      0321508793    <-- For your major

Number of books for the BUS major: 3
```

```
Enter your major: COS
COS221      Programming with C++      Y. Daniel Liang      0136097200    <-- For your major
COS120      Introduction to Programming David Schneider      013212856X    <-- For your major
Tradebook   Full Stack Applications      Stephen Prata      1571691316
BUS103      Exploring Microsoft Access      Robert Grauer      0135098599
COS115      Blended HTML, XHTML, CSS      Julie Bojak      missing    <-- For your major
Tradebook   Project Management      Mike Dawson      1423902270
COS220      Absolute C++      Walter Savitch      0201709279    <-- For your major
Tradebook   Computer Hardware Essentials      Donna Gosselin      missing
BUS200      Database SQL Queries      Aimee Heldon      0618016902
BUS240      Professional Excel      Rob Bovey      0321508793

Number of books for the COS major: 4
```

Need Help?

1. Email your question with your attached .cpp source code file. Do not attach an image or a pdf. Do not paste your code into the email body. I would like to download your code so I can test it as needed.
2. Use the scheduling software on the left side of Brightspace to schedule a Zoom meeting with me. If you cannot make the listed times, email me a list of your free times.