View->Web Layout – For the testCases to appear better

EN

# 1.

Create a class Book that stores information about:

* Name of the book (dynamically allocated array of characters)
* Name of the author (dynamically allocated array of characters)
* The year of release (int)

For this class implement the nessecary constructors and destructor.

Implement the printing operator << and the reading operator >>.

The printing should be done in the format:

Name Author Year

Additionaly implement other methods depending on the needs of your code.

Then create a class Library that stores information about:

* Name of the library (dynamically allocated array of characters)
* An array of objects of class book (Book)
* The number of books (int)
* A variable that indicates a shelf (the space for books) - (int)

For this class implement the nessecary constructors and destructor.

Implement the printing operator <<. The printing should be done in the format:

LibraryName

Book1

Book2

...

Also implement the operator += to add a new book in to the library.

The adding should be done such as when the number of books is zero the value of shelf size should be updated to 5. When the number of books is equal to the size of the shelf you double the size of the shelf and throw an Exception class LibraryFull which prints out the message “Library is full”. So when the number of books is equal to the size of the shelf the new book that is going to be added will be ignored.

For example the number of books reached the shelf size with 5 books, now when we try to add the 6th book we throw the exception and we don’t add the 6th book in our Library.

In addition we implement a method bookByAuthor wich accepts the name of the author and reference to integer and returs an array of books which were written by that author. The reference to integer should be used as a counter for our found books.

# 2.

Create an abstract class Post that stores information about:

* The username (array of 30 characters)
* The content of the post (dynamically allocated array of characters)
* The number of likes (int)
* The number of comments (int)
* A Boolean variable that show if the post has photo or not

This class should have two pure virtual functions:

* double popularity()
* void print()

From this class implement two separate classes FacebookPost and TwitterPost.

For the FacebookPost additionally keep information about the number of shares. For the TwitterPost additionally keep information about the number of retweets and replies.

For the FacebookPost the popularity method should be implemented such as, if the number of shares is greater than 20 the popularity increases by 15%. If the post has photo the popularity gets increased by additionaly 10%. The default value of the popularity is likes \*comments.

For the TwitterPost the popularity method should be implemented shuch as, if the number of replies is greater than 10 the popularity increases by 10%, if the number of retweets is greater than 10 the popularity increases by another 10%, if the post has photo by another 10% and if the content contains the phrase ‘#crypto’ the popularity increases by additionaly 20%.

You can see the format of printing by looking the test case below.

Implement the operator > which compares two post by their popularity.

In addition you should implement a global function double highestPopularity(Post \*\*posts, int n) which accepts an array of pointers of class Post and the size of the array and returns the value of the highest popularity in that array.