COS2614 Assignment 1 : Building a Console-Based Library Management System
Total Marks: 100

Objective:

This assignment aims to reinforce your understanding of Object-Oriented Programming (OOP) principles in C++ by developing a console-based Library Management System. You will implement key OOP concepts such as encapsulation, inheritance, polymorphism, and abstraction.

Assignment Description:

You are required to develop a simple Library Management System using C++. The system should allow users to perform basic operations such as adding books, searching for books, borrowing books, and returning books, all via a console interface.

Requirements:

1. **Class Design (OOP Concepts) - 40 Marks**
    - Create a base class `LibraryItem` that contains attributes `title`, `author`, and `id` and isBorrowed.
    - Create derived classes `Book` and `Magazine` that inherit from `LibraryItem` and add relevant attributes (`genre` for `Book`, `issueNumber` for `Magazine`).
    - Implement encapsulation by making member variables private and using getter and setter methods.
    - Implement polymorphism by creating a virtual function `displayInfo()` in the base class and overriding it in derived classes.
2. **Console Interface - 30 Marks**
    - Implement a menu-driven interface that allows users to interact with the system.
    - Provide options to add books, search for books, borrow books, and return books.
    - Display available books and magazines in a formatted manner.
3. **File Handling for Data Persistence - 20 Marks**
    - Store book and magazine data in a file (e.g., `library_data.txt`).
    - Load the data when the application starts.
    - Save new entries when a book or magazine is added.
4. **Implementation of Borrowing and Returning Feature - 20 Marks**
    - Implement a function to mark a book/magazine as borrowed.
    - Prevent borrowing an already borrowed book.
    - Allow returning a borrowed book and update the status accordingly.

Submission Guidelines:

1. Submit a compressed `.zip` file containing:
    - Source code (`.cpp` and `.h` files)
    - Any required resource files
    - A README file with setup instructions
2. Ensure the code is well-documented with comments.
3. The program should compile and run without errors.

Grading Criteria:

Criteria	Marks
Class Design (OOP)	40
Console Interface	30
File Handling	20
Borrow/Return Functionality	20
Total	**100**