1. Create a class `BookAccount` that represents a library book account. The class should have the following attributes and methods:

   Attributes:

   - ○ `bookID` (public): A string representing the unique identifier for the book.
   - ○ `borrowerName` (public): A string representing the name of the borrower.
   - ○ `fine` (private): A double representing the fine for overdue days.

   Constructor:

   - ○ A constructor that initializes `bookID`, `borrowerName`, and calculates the fine based on the number of overdue days. The fine is calculated as $0.50 per day.

   Destructor:

   - ○ A destructor that displays a message when the account is closed and prints the attributes of the account.

   Methods:

   - ○ A method `calculateFine(int overdueDays)` that calculates the fine based on the number of overdue days and sets the `fine` attribute.
   - ○ A helper method `calculateOverdueDays(int borrowDate, int currentDate)` that calculates the number of overdue days based on the borrow date and current date, represented as integers in the format `YYYYMMDD`. You will need to use this function to get the overdue days.

N.B. [Use the helper function `calculateOverdueDays(int borrowDate, int currentDate)` to calculate overdue days, don't pass the days directly to calculateFine() function. You can take the date in integer format like (YYYYMMDD) or whichever format you prefer. But focus on the logic in this method.]

2. Design a class Room with private attributes length, width, and height.

- The constructor should set the values of these attributes either by user input or with default dimensions (length = 12, width = 8, height = 10).
- Create Room objects: one with default dimensions , one with user input , one with passing parameters directly from code .

- Try creating an object pointer that would point to one of the existing objects.
- Try creating new objects with a pointer. (With default values +
  with parameterized values + with copied value from an existing one)
- Implement a method to calculate and display the floor area of the room (length × width)
- Implement another non member method which will take 2 rooms as parameters, and return the room with larger volume. Hint: You can do this in 2 ways , either friend function or using getter and setter methods.

3. Create a class LibraryBook with private attributes bookTitle, borrowerName, and borrowDays.

- Create an array of 20 LibraryBook objects.
- Write a function to **take input details** (**from user**) for each book (book title, borrower name, and days borrowed).
- Take a 2D array of 2X2 dimension and try setting the attributes and displaying the outputs.
- Implement a function to display books borrowed for more than a specific number of days (e.g., more than 30 days).
- Write another function to display the total number of books borrowed by a specific borrower.

4. You are creating a system where a streaming platform can access and display a private rating for a movie. The Movie class contains private attributes title (string) and rating (float), while the StreamingPlatform class has a method displayRating() that is a friend of the Movie class. This method allows the platform to access and display the private rating of the movie.

5. Create two classes, `Person` and `Address`. The `Person` class should contain information about a person's name and age, while the `Address` class should contain details about a person's address, including street, city, and postal code.

1. **Classes**:
   - **Person**:
     - Attributes: `name` (string), `age` (int).
     - Methods: Constructor to initialize attributes.
   - **Address**:

        - Attributes: `street` (string), `city` (string), `postalCode` (string).
        - Methods: Constructor to initialize attributes.

2. **Friend Function**:
    - Create a friend function `displayDetails` that takes a `Person` object and an `Address` object as parameters and displays the complete details of the person, including their address.