Lab 04 Tasks

Task 01:

Construct a class representing a `Book`, encompassing attributes like `name`, `author`, `ISBN number`, and `total page count`, along with a variable tracking the `number of pages` read. Tasks include defining constructors for the `Book class` - one default constructor and another parameterized constructor initializing all attributes. Values can be hardcoded within the `main` function. Additionally, implement a method to update the `pages read`, triggering a display of "`You have finished the book`" when all pages are read. This method should be called from the `main` function to monitor reading progress.

Task 02:

Revise the previous `Book` class by eliminating the parameterized constructor and enhancing the default constructor. The revised default constructor should initialize the `page count to 1000` and the `pages read` to `0` using a member initialization list. Generate `setter` methods for the remaining attributes and utilize them within the `main` function to update attribute values. Introduce a method called "`showBookInfo`" to exhibit all book details. Invoke this method from the `main` function for displaying book information.

Task 03:

Develop a class called `WeekDays` containing `private` data members - a `string` array named `Days` for the days of the week and an integer variable named `CurrentDay` for the current day. Create the following constructors and member functions for the class: a Default Constructor initializing the `Days` array from Sunday to Saturday, a Parameterized Constructor accepting an integer value for `CurrentDay`, performing a modulus operation to adjust if the value exceeds 6, and initializing the `Days` array accordingly. Additional member functions include `getCurrentDay`, `getNextDay`, `getPreviousDay`, and `getNthDayFromToday`.

Task 04:

Instantiate a class named `Office` to represent attributes of an office space, including `Location` (default value: " "), `SeatingCapacity` (default value: 0), and `Furniture` (a `string` array with a size of `3`, default: {"", "", ""}). Implement a parameterized constructor assigning default values to all attributes using a member initialization list. Create multiple instances of the Office class using different numbers of constructor arguments.

Task 05:

Transform the `Office` class by replacing the `furniture` array with a pointer. Modify the constructor to accept the size of the `furniture` array instead of the array itself, dynamically allocating memory for the `furniture` array within the constructor. Implement a destructor to deallocate memory allocated for the `furniture` pointer. In the main function, dynamically allocate an `Office` object, pass required values to the constructor, and then `delete` the dynamically allocated object to ensure proper invocation of the destructor.