OBJECT ORIENTED PROGRAMMING – SPRING 2022

Dead line: 30th *April* 2022

Assignment # 3
Total Marks: 60

[1] Create a class called time that has separate int member data for hours, minutes, and seconds. One constructor should initialize this data to 0, and another should initialize it to fixed values. Another member function should display it, in 11:59:59 format.

[2] Modify the class time that you have created in Q # 1 to include overloaded increment (++) and decrement (--) operators that operate in both prefix and postfix notation and return values. Add statements to main() to test these operators.

[3] Add to the time class of Q # 1 the ability to subtract two time values using the overloaded (-) operator, and to multiply a time value by a number of type float, using the overloaded (*) operator.

[4] Imagine a publishing company that markets both book and audiocassette versions of its works. Create a class publication that stores the title (a string) and price (type float) of a publication. From this class derive two classes: book, which adds a page count (type int), and tape, which adds a playing time in minutes (type float). Each of these three classes should have a getdata() function to get its data from the user at the keyboard, and a putdata() function to display its data. Write a main() program to test the book and tape classes by creating instances of them, asking the user to fill in data with getdata(), and then displaying the data with putdata().

[5] Start with the publication, book, and tape classes of Exercise 3. Add a base class sales that holds an array of three floats so that it can record the dollar sales of a particular publication for the last three months. Include a getdata() function to get three sales amounts from the user, and a putdata() function to display the sales figures. Alter the book and tape classes so they are derived from both publication and sales. An object of class book or tape should input and output sales data along with its other data. Write a main() function to create a book object and a tape object and exercise their input/output capabilities.

[6] Assume that the publisher in Exercises 4 and 5 decides to add a third way to distribute books: on computer disk, for those who like to do their reading on their laptop. Add a disk class that, like book and tape, is derived from publication. The disk class should incorporate the same member functions as the other classes. The data item unique to this class is the disk type: either CD or DVD.