



# CS217 - Object-oriented Programming (OOP)

## Assignment # 2

**Max Points:** 20

**Due Date:** Sunday, April 25, 2021, 11 p.m.

### Carefully read the following instructions!

- It should be clear that your assignment would not get any credit if the assignment is submitted after the due date. No assignment will be accepted after the due date.
- Strict action will be taken if submitted solution is copied from any other student.
- If you find any confusion in assignment (Question statement), please consult before the deadline. After the deadline no queries will be entertained in this regard.
- For any query, feel free to email at: **basit.jasani@nu.edu.pk**
- **Submission:** Submission will only be accepted through GOOGLE CLASSROOM. Submit all your codes in a single folder name it as your Student ID “**KXX-XXXX**”. The folder will contain five text files as ***Q1.txt, Q2-Task-A.txt, Q2-Task-B.txt, Q2-Task-C.txt and Q2-Task-D.txt*** with proper commenting of the code.
- **CAUTION:** Seven students got ZERO marks in A#1 due to **PLAGIARISM.**

---

### Question # 1

Package-delivery services, such as FedEx®, DHL® and UPS®, offer a number of different shipping options, each with specific costs associated. Create an inheritance hierarchy to represent various types of packages. Use Package as the base class of the hierarchy, then include classes TwoDayPackage and OvernightPackage that derive from Package. Base class Package should include data members representing the name, address, city, state and ZIP code for both the sender and the recipient of the package, in addition to data members that store the weight (in ounces) and cost per ounce to ship the package. Package's constructor should initialize these data members. Ensure that the weight and cost per ounce contain positive values. Package should provide a public member function calculateCost that returns a double indicating the cost associated with shipping the package. Package's calculateCost function should determine the cost by multiplying the weight by the cost per ounce. Derived class TwoDayPackage should inherit the functionality of base class Package, but also include a data member that represents a flat fee that the shipping company charges for two-day-delivery service. TwoDayPackage's constructor should receive a value to initialize this data member. TwoDayPackage should redefine member function calculateCost so that it computes the shipping cost by adding the flat fee to the weight-based cost calculated by base class Package's calculateCost function. Class OvernightPackage should inherit directly from class Package and contain an additional data member representing an additional fee per ounce charged for overnight-delivery service. OvernightPackage should redefine member function calculateCost so that it adds the additional fee per ounce to the standard cost per ounce before calculating the shipping cost. Write a test program that creates objects of each type of Package and tests member function calculateCost.

---

### Question # 2

#### Task a)

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 through console, 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().

#### Task b)

Start with the publication, book, and tape classes of “Task A”. 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.

### **Task c)**

Assume that the publisher in “**Task A and B**” 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.

### **Task d)**

Suppose you want to add the date of publication for both books and tapes. From the publication class, derive a new class called publication2 that includes this member data. Then change book and tape so they are derived from publication2 instead of publication. Make all the necessary changes in member functions so the user can input and output dates along with the other data.

**XXXXX Happy Coding XXXXX**