Homework # 4

Create a class called "box" that has the following attributes (variables): length, width, height (in inches), weight (in pounds), address (1 line), city, state, zip code. These variables must be **private**. Create **setter and getter functions** for each of these variables. Also create **two constructors** for the box class, one default constructor that takes no parameters and sets everything to default values, and one which takes parameters for all of the above. Create a **calcShippingPrice** function that returns the cost of shipping a box, using the following formula:

Shipping price for a single box = (((length + width + height) * $0.50) + (weight * $1.00))

Finally, create a **print function** that prints length, width, height, address, city, state, zip code and shipping price to the screen.

You may not end up needing all of the above functions in main but you still need to create and test them all so that your box class is versatile and can be used by others.

Main should create an array of 3 boxes. Have the user enter the information for each box, then display the **information for all boxes** as well as the **total shipping price for all boxes combined**.

You must use a class file, header file, and main file (3 files with code in them).

Show test run(s) proving that your program works.

Input validation:

Length, width, height, weight should all be positive. If negative or default constructor is used set to 0.

Address needs no input validation, but can have spaces in it (remember getline and ignore), if default constructor is used set to blank (assume all addresses are 1 line only).

City needs no input validation but can have spaces in it, if default constructor is used set to blank.

State should be exactly two characters long. If invalid or default constructor is used set to "XX" (you do NOT need to check if the state is "real", i.e. "YZ" would be valid input even though there is no state with that abbreviation – Hint: string.length()).

Zip code should be 5 digits and positive (no leading zeros), if invalid or default constructor is used set to 0.

Input validation should be done in your setter functions and/or constructors as needed to **ensure no bad data can get in to the class variables.** Invalid input should instead set the value to a default as specified above.

Your program should be **split into three files**, a .h file for the box class, a .cpp file for the box class, and a .cpp file for main.

Your repl must include the following:
- Your code
- Screenshot(s) of test run(s) of the program showing it being thoroughly tested.
- A flowchart and/or written algorithm showing the design of your code.
Make sure nothing in your code results in junk values or causes a segmentation fault.