Object Oriented Programming Lab

Lab 13 **Marks 10**

Instructions

Work on this lab individually. You can use your books, notes, handouts etc. but you are not allowed to borrow anything from your peer student. *You are strictly **NOT ALLOWED** to include any additional data-members/functions/constructors* in your class.

Marking Criteria

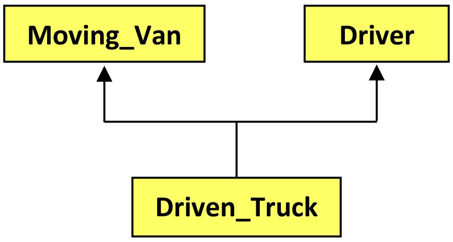
Show your work to the instructor before leaving the lab to get some or full credit.

What you must do

Program the following task in your C++ compiler and then compile and execute them. *Write the **main** function first and keep testing the functionality of each function once created.*

Vehicle Inheritance

Implement the following class hierarchy, the inheritance access level should be **public** for **Driven_Truck** class.



1. Add protected data members named **payload**, **weight,** and **mpg** to **Moving_Van** class all of **float** type.
2. Add a member function named **initialize** that initializes all the data members of **Moving_Van** class with the values passed to it as arguments.
3. Add a member function named **efficiency** which calculates and return the efficiency of a particular **Moving_Van** object, the following formula can be used to calculate the efficiency: $payload\ /\ (payload\ +\ weight)$
4. Add a member function named **cost_per_ton** the function accepts a parameter named **fuel_cost** of **float** type then calculates and return the cost/ton with the help of following formula: $fuel\_cost\ /\ (payload\ /\ 2000.0)$
5. Add protected data members named **hourly_pay** and **weight** to **Driver** class all of **float** type.
6. Add a member function named **initialize** that initializes all the data members of **Driver** class with the values passed to it.
7. Add a member function named **cost_per_mile** the which calculates and return the cost/mile for a particular **Driver** class object, the following formula can be used to calculate the cost/mile: $hourly\_pay\ /\ 55.0$
8. Add a member function named **drivers_weight** that simply returns the **weight** of a particular **Driver** object.
9. Add a member function named **initialize** that initializes all the data members of **Driven_Truck** class (inherited from its base classes) with the values passed to it. *You are not allowed to make a call to any of the **initialize** methods of your base classes (Moving_Van or Driver)*.
10. Add a member function named **cost_per_full_day** to **Driven_Truck** class that accepts a parameter named **cost_of_gas** then calculates and return the cost/day for a particular object of a **Driven_Truck**, the cost can be calculated with the help of following formula: $(8.0\ *\ hourly\_pay\ +\ 8.0\ *\ cost\_of\_gas\ *\ 55.0)\ /\ mpg$
11. Add a member function named **total_weight** which calculates and return the total weight, which can be calculated with the addition of **Moving_Van** and **Driver's** weight.
12. Write down your main method and create an object named **chuck_ford** of **Driven_Truck** class and perform the following operations on it
    1. Make a call to **initialize** method of your **Driven_Truck** class.
    2. Make a call to **initialize** method of your Driver.
    3. Calculate and display the **efficiency** of your **chuck_ford** object.
    4. Calculate and display the **cost/mile** of your **chuck_ford** object.
    5. Calculate and display the **cost/day** of your **chuck_ford** object.
    6. Calculate and display the **total weight** of your **chuck_ford** object.

Now add **default**, **parameterized** and **copy constructors** in each of the created class above. Test and observe the constructor call sequences of your newly written code by creating different objects with the help of default, parameterized and copy constructors.

☺ ☺ ☺ **BEST OF LUCK** ☺ ☺ ☺