COMP 2404
Introduction to Software Engineering
Assignment 5

Released: Monday March 26, 2018, 12:00 Noon

Due: Wednesday April 11, 2018, 12:00 Noon

Instructions

In this assignment you will build upon the program worked on in the previous assignment. You are free to build upon your solution or start from the skeleton code provided.

Task 1: Linked List Template Class

Your first task is to replace the various Linked List collection classes in the code with a single **LinkedList** Template class that will be able to store any objects. As was done in class and your tutorials, you will create a LinkedList.h file that contains the class definition as well as the function definitions. Once implemented this template will be used in place of the **CustomerList** and **VehicleList** class (storing Customer and Vehicle pointers, respectively). Your **LinkedList** template must be consistent with the previously used classes (ie. must implement the same member functions and override the same operators). There is one function in the **CustomerList** class that we will not provide in our generic template, the **get** function. The only place this function is called from is the **Shop** class. Modify the **getCustomer** function in the **Shop** class that calls this **get** function so the functionality is preserved (ie. use the provided functions and operators provided by the **LinkedList** template to provide this functionality).

You will also store pointers to the **Mechanic** objects in the **Shop** class in a **LinkedList**. In doing this you will need to look at the following:

- Update how the **+=** operator works for **Mechanic** pointers in the **Shop** class
- Get rid of any data members or member functions that are no longer needed in the **Shop** class
- Update how certain member functions work in light of this change in the **Shop** class
- Update how mechanics are printed in the controller and view classes
- Overload the **<** and **>** operators in the **Mechanic** class to compare objects based on salary. This is needed since the **LinkedList** inserts and stores its objects in sorted order.

Task 2: Vehicle Class Hierarchy

You will now introduce a second inheritance hierarchy into your program. The base of this hierarchy will be the **Vehicle** class. You are free to change the access specifiers in this class as you see fit. You are to add a pure virtual to this class called **toString** that will return a string and update the overloaded output stream insertion operator to call to call the **toString** function as opposed to accessing the member variables directly. This will allow us to control how all derived classes in this hierarchy will be printed to screen (ie. allowing us to include class specific information). See the sample output for an example of this in use.

You will now introduce three derived classes to this hierarchy. The classes and the minimum of what you need to include in them are described below.

- The **Car** class will not introduce any further data members to those supplied by the **Vehicle** class. You will need to:
    - Include an appropriate constructor
    - Implement the **toString** function appropriately and make any other changes so that Car objects can easily be printed to screen.
- The **Truck** class will include everything described for the **Car** class as well as:
    - A new integer data member to keep track of the number of axles the truck has
- The **Motorcycle** class will include everything described for the **Car** class as well as:
    - A new boolean data member to indicate if the motorcycle has a sidecar or not

Task 3: VehicleFactory class

You will now create a **VehicleFactory** class to implement the factory design pattern within your program. This class will offer three public functions, all called **create** which all return a **Vehicle** pointer. They will be differentiated by their parameter list: each version is responsible for creating a specific object in the **Vehicle** hierarchy and the parameters it takes are those needed to create the specific object in question. Once implemented, the header file for this class should be the only file in your entire program (other than the Car.cc, Truck.cc and Motorcycle.cc source code files) that includes the **Car**, **Truck** and **Motorcycle** headers . The rest of your program will interact with these objects using **Vehicle** handles.

Update your **ShopController** class to include a **VehicleFactory** data member. This instance of your factory will be responsible for creating every instance of any object in the **Vehicle** hierarchy throughout your program. Update your **initShop** function in the **ShopController** class to use your factory to create vehicles of all types to your customer database.

Task 4: Update UI

The last task is to update the user interface (and the associated view and controller classes) to take advantage of the new functionality. Specifically, when the user asks to add a vehicle and after prompting the user for the id, you must also prompt the user for which type of vehicle they want to add. Based on that, the appropriate information is asked for, the appropriate object is created (using your new **VehicleFactory**) and is added to that customer. As always, you must maintain the good design principles of this program, including the separation of the view, controller and model objects. See the sample output to see how this should work.

Constraints

- your program must not have any memory leaks (don't worry if valgrind reports that some memory on the heap is "still reachable")
- do not use any global variables
- your program must reuse functions everywhere possible
- your program must be thoroughly commented
- **your program must compile and run in COMP2404-2406- W18 Virtual Machine**

Submission

You will submit in cuLearn, before the due date and time, one **tar** file that includes all the following:

- all source code, including the code provided
- a readme file that includes:
    - a preamble (program author, purpose, list of source/header/data files)
    - the exact compilation command
    - launching and operating instructions

Grading [out of 30 marks]

Marking components:

- 11 marks: linked list template
    - 5 marks: overall structure
    - 1 mark: each for storing **Mechanic**, **Vehicle** and **Customer** objects using template
    - 1 mark: updated **+=** operator for **Mechanic** objects in **Shop** class
    - 1 mark: for other appropriate updates to **Shop** class
    - 1 mark: for overloaded **<** and **>** in **Mechanic** class based on salary
- 10 marks: **Vehicle** class hierarchy
    - 1 mark: for pure virtual **toString** function
    - 3 marks: for each **Car**, **Truck** and **Motorcycle** class structure
- 5 marks: **VehicleFactory** class
    - 3 marks: correctly implements each version of the **create** function
    - 2 marks: proper use in the **ShopController** class
- 4 marks: updates to UI

Notes:
In order to get credit for a marking component, the program must prove that the marking component executes successfully. This is usually accomplished by printing out correct data.

Deductions

- Packaging errors:
  - 10% for missing readme
- Memory leaks:
  - 10% for any memory leaks in your program
- Major programming and design errors:
  - 50% of a marking component that uses global variables
  - 50% of a marking component that consistently fails to use correct design principles, including separate functions
  - 50% of a marking component where unauthorized changes have been made to provided code or prototypes
- Minor programming errors:
  - 10% for consistently missing comments or other bad style
  - 10% for consistently failing to perform basic error checking
- Execution errors:
  - 100% of a marking component that can't be tested because the code doesn't compile or execute in the VM
  - 100% of a marking component that can't be tested because the feature isn't used in the code
  - 100% of a marking component that can't be proven to run successfully because data is not printed out

Sample Console Output

User input highlighted.

```
> make clean
rm -f main.o ShopController.o View.o VehicleFactory.o Shop.o Customer.o Vehicle.o
Car.o Truck.o Motorcycle.o Mechanic.o Person.o mechanicshop
> make
g++ -c main.cc
g++ -c ShopController.cc
g++ -c View.cc
g++ -c VehicleFactory.cc
g++ -c Shop.cc
g++ -c Customer.cc
g++ -c Vehicle.cc
g++ -c Car.cc
g++ -c Truck.cc
g++ -c Motorcycle.cc
g++ -c Mechanic.cc
g++ -c Person.cc
g++ -o mechanicshop main.o ShopController.o View.o VehicleFactory.o Shop.o Customer.o
Vehicle.o Car.o Truck.o Motorcycle.o Mechanic.o Person.o
```

```
> ./mechanicshop


          **** Toby's Auto Mechanic Information Management System ****

                                    MAIN MENU

          1. Print Customer Database

          2. Add Customer

          3. Add Vehicle

          4. Remove Customer

          5. Remove Vehicle

          6. Print Mechanics

          0. Exit

Enter your selection:  1

CUSTOMERS:

Customer ID 1001

    Name:                           Abigail Atwood
    Address:                        43 Carling Dr.
    Phone Number:                    (613)345-6743

    1 vehicle(s):

       CAR:          Green 2016   Subaru Forester (40000km)


Customer ID 1002

    Name:                            Brook Banding
    Address:                        1 Bayshore Dr.
    Phone Number:                    (613)123-7456

    2 vehicle(s):

       CAR:          White 1972 Volkswagon Beetle (5000km)
       CAR:          White 2018      Honda Accord (5000km)


Customer ID 1004
```

Name: Eve Engram
Address: 75 Bronson Ave.
Phone Number: (613)456-2345

5 vehicle(s):

CAR:	Green	2013	Toyota Corolla (80000km)
MOTORCYCLE:	Grey	2014	Kawasaki Z650 (5000km), doesn't have sidecar
CAR:	Gold	2015	Toyota Rav4 (20000km)
CAR:	Blue	2017	Toyota Prius (10000km)
MOTORCYCLE:	Blue	2018	Kawasaki Ninja (100km), doesn't have sidecar

Customer ID 1003

Name: Ethan Esser
Address: 245 Rideau St.
Phone Number: (613)234-9677

2 vehicle(s):

TRUCK:	Red	2000	Mack Granite (500000km), 8 axles
CAR:	Black	2010	Toyota Camery (50000km)

Customer ID 1000

Name: Maurice Mooney
Address: 2600 Colonel By Dr.
Phone Number: (613)728-9568

2 vehicle(s):

CAR:	Red	2007	Ford Fiesta (100000km)
TRUCK:	Black	2017	Ford F150 (1000km), 3 axles

Customer ID 1005

Name: Victor Vanvalkenburg
Address: 425 O'Connor St.
Phone Number: (613)432-7622

5 vehicle(s):

MOTORCYCLE:	Black	2006	Harley Touring (8000km), has sidecar
CAR:	Purple	2012	GM Envoy (60000km)
CAR:	Orange	2012	GM Trailblazer (90000km)
CAR:	Red	2015	GM Malibu (20000km)

```
       CAR:          Black 2016       GM Escalade (40000km)


Press enter to continue...


          **** Toby's Auto Mechanic Information Management System ****

                                MAIN MENU

          1. Print Customer Database

          2. Add Customer

          3. Add Vehicle

          4. Remove Customer

          5. Remove Vehicle

          6. Print Mechanics

          0. Exit

Enter your selection:  3
Customer ID: 1000

Vehicle Type (1: Car, 2: Truck, 3: Motorcycle): 2

Make: Toyota
Model: Tacoma
Colour: Blue
Year: 2018
Mileage: 1000
Number of axles: 2

Press enter to continue...


          **** Toby's Auto Mechanic Information Management System ****

                                MAIN MENU

          1. Print Customer Database

          2. Add Customer
```

```
        3. Add Vehicle

        4. Remove Customer

        5. Remove Vehicle

        6. Print Mechanics

        0. Exit

Enter your selection:  3
Customer ID: 1002

Vehicle Type (1: Car, 2: Truck, 3: Motorcycle): 3

Make: Suzuki
Model: Cruiser
Colour: Black
Year: 2000
Mileage: 50000
Has sidecar (y/n): y

Press enter to continue...



         **** Toby's Auto Mechanic Information Management System ****

                                MAIN MENU

        1. Print Customer Database

        2. Add Customer

        3. Add Vehicle

        4. Remove Customer

        5. Remove Vehicle

        6. Print Mechanics

        0. Exit

Enter your selection:  1

CUSTOMERS:

Customer ID 1001
```

Name: Abigail Atwood
Address: 43 Carling Dr.
Phone Number: (613)345-6743

1 vehicle(s):

CAR: Green 2016 Subaru Forester (40000km)

Customer ID 1002

Name: Brook Banding
Address: 1 Bayshore Dr.
Phone Number: (613)123-7456

3 vehicle(s):

CAR: White 1972 Volkswagon Beetle (5000km)
MOTORCYCLE: Black 2000 Suzuki Cruiser (50000km), has sidecar
CAR: White 2018 Honda Accord (5000km)

Customer ID 1004

Name: Eve Engram
Address: 75 Bronson Ave.
Phone Number: (613)456-2345

5 vehicle(s):

CAR: Green 2013 Toyota Corolla (80000km)
MOTORCYCLE: Grey 2014 Kawasaki Z650 (5000km), doesn't have sidecar
CAR: Gold 2015 Toyota Rav4 (20000km)
CAR: Blue 2017 Toyota Prius (10000km)
MOTORCYCLE: Blue 2018 Kawasaki Ninja (100km), doesn't have sidecar

Customer ID 1003

Name: Ethan Esser
Address: 245 Rideau St.
Phone Number: (613)234-9677

2 vehicle(s):

TRUCK: Red 2000 Mack Granite (500000km), 8 axles
CAR: Black 2010 Toyota Camery (50000km)

Customer ID 1000

```
    Name:                           Maurice Mooney
    Address:                  2600 Colonel By Dr.
    Phone Number:                    (613)728-9568

    3 vehicle(s):

       CAR:            Red 2007       Ford Fiesta (100000km)
       TRUCK:        Black 2017         Ford F150 (1000km), 3 axles
       TRUCK:         Blue 2018     Toyota Tacoma (1000km), 2 axles


Customer ID 1005

    Name:                     Victor Vanvalkenburg
    Address:                      425 O'Connor St.
    Phone Number:                    (613)432-7622

    5 vehicle(s):

       MOTORCYCLE:   Black 2006    Harley Touring (8000km), has sidecar
       CAR:         Purple 2012          GM Envoy (60000km)
       CAR:         Orange 2012    GM Trailblazer (90000km)
       CAR:            Red 2015         GM Malibu (20000km)
       CAR:          Black 2016       GM Escalade (40000km)



Press enter to continue...



         **** Toby's Auto Mechanic Information Management System ****

                                MAIN MENU

        1. Print Customer Database

        2. Add Customer

        3. Add Vehicle

        4. Remove Customer

        5. Remove Vehicle

        6. Print Mechanics

        0. Exit
```

```
Enter your selection:  6

MECHANICS:

Employee ID 5001

    Name:                              Steve Bane
    Address:                           77 Oak St.
    Phone Number:                    (613)223-4653
    Salary:                                 60000
Employee ID 5002

    Name:                              Jane Smyth
    Address:                         10 5th Ave.
    Phone Number:                    (613)762-4678
    Salary:                                 71000
Employee ID 5000

    Name:                             Bill Taylor
    Address:                        54 Park Place
    Phone Number:                    (613)826-9847
    Salary:                                 75000

Press enter to continue...



          **** Toby's Auto Mechanic Information Management System ****

                                  MAIN MENU

         1. Print Customer Database

         2. Add Customer

         3. Add Vehicle

         4. Remove Customer

         5. Remove Vehicle

         6. Print Mechanics

         0. Exit

Enter your selection:  0

>
```