Name: Jingmin Zhou (Jessica)

Student ID: 119766194

Email: [jzhou175@myseneca.ca](mailto:jzhou175@myseneca.ca)

In the Workshop 7 (Derived Classes & Custom I/O Operators), I have learned how to work with classes that make up hierarchal structure. I have learned to inherit a derived class (Car) from a base class (Vehicle), shadow a base class member function with a derived class member function (e.g. display()), access a shadowed member function that is defined in a base class, like Vehicle display() called in Car class. Additionally, custom input/output operators are utilized with Vehicle and Car class.

Firstly, derived class Car is a kind of based class Vehicle, which demonstrates the inheritance relationship between two classes. Except its own instance variables and member functions, a derived class contains all of the instance variables and all of the normal member functions of its base class, but it does not inherit constructors, destructors or assignment operators of based class.

Secondly, a member function of a derived class shadows the base class member function with the same identifier, like functions finetune(), display(ostream& os) and input(istream& in) in this workshop. To access the base class version of a member function that a derived class version has shadowed, we use scope resolution. For example, Vehicle is used to identify the shadowed function and access it in Car class.

Finally, input and output functions are utilized and the display(ostream& os) function is defined in two classes. The Vehicles **display** function is called in Car class. It is supposed to think about how to call it and display the information as required format.