Ryan Hoffman
CS 202, Lab#: 1107
Project 6 Documentation

Compile Command: ***1.)*** *make all* ***2.)*** *./proj6*

```
Car c1;
```

-Creates a car object using the default constructor. Sets throttle to 0. Prints debug message.

```
float lla_rno[3] = {39.54, 119.82, 4500.0};
Car c_rno(lla_rno);
```

-Creates car object using the parameterized constructor. Sets throttle to 0. Sets passed LLA values. Prints debug message.

```
Car c_cpy( c_rno );
```

-Creates a new car object using the copy constructor. Sets car values from c_rno object. Prints debug message.

```
c1 = c_cpy;
```

-Calls assignment operator overload and sets values from c_cpy to c1. Prints debug message.

```
float lla_new[3] = {37.77, 122.42, 52.0};
c1.Move( lla_new );
```

-Calls c1's move function. Prints debug message. Sets drive to 75. Sets LLA values to the passed values.

```
cout << c_rno << endl;
```

-Calls insertion operator overload for c_rno. The insertion operator overload calls the serialize function for the car object and prints the throttle and LLA values.

```
float lla_ny[3] = {40.71, 74.00, 10.0};
c1.SetLLA( lla_ny );
float lla_la[3] = {34.05, 118.24, 71.01};
c_cpy.SetLLA( lla_la );
```

-Set's LLA values for c1 and c_cpy to passed parameter values.

```
Vehicle* vehicles_array[3];
vehicles_array[0] = &c1;
vehicles_array[1] = &c_rno;
vehicles_array[2] = &c_cpy;
```

-Creates an array of vehicle object pointers that point to different car objects.

```
for (int i=0; i<3; ++i){
   vehicles_array[i]->Move( lla_new );
}
```

-For loop that goes through the vehicle pointer array and calls the move function for each car object in the array. The move function prints the debug message, sets the throttle to 75, and sets the LLA values to the parameter values.

```
for (int i=0; i<3; ++i){
  cout << *vehicles_array[i] << endl;
}
```

-For loop that goes through vehicle pointer array and calls the insertion operator overload for each object. The insertion operator overload calls the virtual car serialize function which prints the throttle and LLA values.

The purpose of this program was to continue to practice creating overloaded and overridden functions using inheritance. We also test virtual functions and polymorphism of objects. The only design problem I encountered was that I had forgotten to make the insertion operator overload function not a part of the Vehicle class.