Name: Jingmin Zhou (Jessica)

Student ID: 119766194

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

In the Workshop 5 (Operator Overloading), I have learned how to code a compositional relationship across multiple classes, like define Engine type of array in Ship class, overload an operator as a member of a class using keyword friend, like friend bool operator==(const Ship& ship1, const Ship& ship2) in Ship class, overload an operator as a helper function of a class through coding a global function, and allocate dynamic memory and deallocate memory as needed.

Firstly, when coding operator+= function, it is supposed to dynamically create a new Engine array of original number of engines plus 1, copy the old array's contents into the new one, add the object of type engine to the array, deallocate the memory of the old Engine array to avoid memory leak.

Secondly, a helper operator is a global function that overload operator, like helper operator< in this workshop, which only have access to the public member functions in class Ship. It is different with friend function operator==, which is granted with access to the private members of the Ship class, not only the public member functions.

Finally, an array of Engine objects is include in Ship class, which means Ship inherits the structure of Engine and is an important object-oriented concept in OOP. Inheritance of object-oriented programming allow us to utilize existing technology, like the structure functions of Engine reused in Ship class.