// OOP345 Workshop 6: STL Containers

// File: reflect.txt

// Name: Jingmin (Jessica) Zhou

// Email: jzhou175@myseneca.ca

// ID: 119766194

// Date: 07/09/2020

In Workshop 6 (STL Containers), I have learned how to manage polymorphic objects (e.g. Car and Racecar) using the vector container of the STL, move a dynamically allocated object into the vector container, code a range-based iteration on the objects in a container and report and handle an exception (e.g. invalid record or invalid data type).

Firstly, the pointer (Vehicle\*) in the vector (m\_vehicles) point to dynamically allocated memory for instances of Car or Racecar in function createInstance(std::istream& in). Therefore, it is needed to delete the dynamically allocated memory in the vector since the vector will not do it automatically. If the objects are stored in a vector, the vector will allocate memory to store them and will deallocate it when the vector destructor runs.

Secondly, when an unknown type of vehicle (not c, C, r, or R) is reported in createInstance(std::istream& in), it will generate and throw an exception with wrong vehicle type ‘tag’. When invalid data member (e.g. condition or topSpeed) is reported in Car(std::istream& in), it will generate and throw an exception with wrong record “Invalid record”. Function loadData() will catch “Unorgnized vehicle type” and “Invalid record” when exception is thrown. Another solution is checking the condition to inform the client of any invalid record by printing the error message. For example, if type of Vehicle tag is not c, C, r, or R, then print the message with “Unrecognized record type: [tag]”.

Thirdly, the element type (<sdds::Vehicle\*>) in vector and list cannot be replaced by <sdds::Vehicle>, because the Vehicle is an abstract class (or interface), which cannot be instantiated. The pointer or references can be used for abstract class (Vehicle) and its specific classes (Car or Racecar).

===========================================================================================================================================

NO Quiz for this week.