BTP305 NCC Workshop 6 Reflection

Sophia Huynh

118448208

[Shuynh9@myseneca.ca](mailto:Shuynh9@myseneca.ca)

In the workshop, we used std::list<sdds::Vehicle\*> and std::vector<sdds::Vehicle\*>, allowing us to use Vehicle dynamically. If we were to make them inline (std::list<sdds::Vehicle> and std::vector<sdds::Vehicle>), we would be able to achieve the same results, just less efficiently. For example: if we ever were to resize the vector, the compiler will have to make a copy of each data item. This is also the reason we need to deallocate the dynamically allocated memory in the vector because if we didn’t, there will be a memory leak for Vehicle objects. loadData() helps to detect what went wrong when building an object because it makes sure the file is opened and read properly and then checks to make sure that the data is correct (ex: recognizing valid vehicles). Using a throw/catch statement, the compiler is able to make sure everything is valid and if not, to throw a error message.