Name: Tian Zheng

Student number: 155394174

Email: tzheng26@myseneca.ca

1. The reason for using the vector container rather than any other available in the STL.

Because the elements in the container are in sequential order. In addition, the size is unknown in advance, so I cannot use std::array. Plus, there will not be any need adding or removing elements frequently at the front or back, or in the middle of the sequence, so I do not neet to use std::deque or std::list. To sum up, std::vector is by default the good top choice as an alternative to built-in arrays.

2. Why there is no need to deallocate the dynamically allocated memory in your own code.

Because vectors are automatically deallocated from heap memory.

3. How the range-based for simplifies coding in this case.

for (auto& e:products) {

e->display(os);

os << std::endl;

total += e->price();

}

By using the range-based for loop, I do not need to declare an iterator to loop through the vector container.

4. I have also learned to use enumeration defined with a class, to decide the corresponding value based on the enumeration constant.

class TaxableProduct : public Product {

public:

enum class taxType { HST, PST, NON };

static void setRate(taxType t);

//other member functions

private:

taxType myType;

static double taxRate;

};

void TaxableProduct::setRate(taxType t) {

switch (t) {

case TaxableProduct::taxType::HST:

taxRate = 0.13;

break;

case TaxableProduct::taxType::PST:

taxRate = 0.08;

break;

default:

taxRate = 0;

}

}