Gebze Technical University
Department of Computer Engineering
CSE 241/505
Object Oriented Programming
Fall 2019
Homework # 6
Inheritance, Templates, STL
Due date
Dec 24th 2019

In this homework, you will write a templated class hierarchy for a simple container class hierarchy.

The class **GTUContainer** is an abstract class with the following pure virtual member functions.

empty	Test whether container is empty
size	Return container size
max_size	Return maximum size
insert	Insert element, throws exception std::bad_pafram if there is a problem with insersion
erase	Erase element
clear	Clear all content
begin	Return iterator to beginning
end	Return iterator to end

The class **GTUSet**<T> derives from the base class and implements all of the functions appropriately for a set class.
The class **GTUVector**<T> derives from the base class and implements all of the functions appropriately for a vector class. This class will also overload the [] operator. You may write other helper classes to make your work easier.

All classes will keep their data using dynamic memory techniques with shared_ptr STL pointers. Do not use regular pointers or STL container classes.

The classes **GTUIterator** and **GTUIteratorConst** implement iterator operators such as *, ->, ++, --, =, and ==.

You will also implement the following global functions from the STL library which will accept GTUIterator parameters instead of regular STL iterators

- **`find`**
- **`find_if`**
- **`for_each`**

Write your driver program to test the all the classes and all of their functions. Do not forget to test the global functions with all concrete classes.

Notes

- Use separate header and implementation files for each class.
- Use name spaces.
- Do not forget to test the thrown exceptions.