

**Gebze Technical University**  
**Department of Computer Engineering**  
**CSE 241/505**  
**Object Oriented Programming**  
**Fall 2017**  
**Homework # 7**  
**Java Inheritance and Generics**  
**Due date**  
**Dec 28<sup>th</sup> 2017**

In this homework, you will write a generic class hierarchy for sets and maps for Java.

GTUSetInt<T> is a Java interface with the following methods.

<b>empty</b>	Test whether container is empty
<b>size</b>	Return container size
<b>max_size</b>	Return maximum size
<b>insert</b>	Insert element, throws exception InvalidParameterException if the element is already in the set
<b>intersection</b>	The returned set is the intersection of this set and the given set.
<b>erase</b>	Erase element
<b>clear</b>	Clear all content
<b>find</b>	Get iterator to element
<b>count</b>	Count elements with a specific value
<b>begin</b>	Return iterator to beginning
<b>end</b>	Return iterator to end

The class GTUSet<T> implements the GTUSetInt interface. It will keep its data using Java arrays. Do not use any Java Collection classes. It implements the intersection method as follows:

**intersection**

```
GTUSetInt<T> intersection(GTUSetInt<T>)
```

The returned set is the intersection of this set and the given set.

The class `GTUMap<K, V>` extends from `GTUSet< javafx.util.Pair <K, V> >` and implements the following extra method

**at**

| Access element

`V at (K k)`

If *k* matches the key of an element in the set, the method returns a reference to its mapped value.

The class `GTUIterator` implements the following methods

boolean	<a href="#"><code>hasNext()</code></a> Returns <code>true</code> if this list iterator has more elements when traversing the list in the forward direction.
boolean	<a href="#"><code>hasPrevious()</code></a> Returns <code>true</code> if this list iterator has more elements when traversing the list in the reverse direction.
<a href="#"><code>T</code></a>	<a href="#"><code>next()</code></a> Returns the next element in the list and advances the cursor position.
<a href="#"><code>T</code></a>	<a href="#"><code>previous()</code></a> Returns the previous element in the list and moves the cursor position backwards.

Write your driver class to test the all the classes and all of their methods.

Notes

- Submit your nicely produced Javadoc documents with your homework.
- Do not forget to test the thrown exceptions