Course: CSCI 221
Term: Spring 2015
Instructor: Paul Buhler

Programming Assignment 6 –
Due Date: 11:59 pm on Tuesday, 04/07
Send you source files via email attachment sent to buhlerp@cofc.edu with the subject line:
csci221 prog 6

This programming assignment is designed to demonstrate a few concepts: pattern oriented software development, interfaces, iterators, and wrapping.

Regarding patterns, this program utilizes two,
the Iterator pattern (http://en.wikipedia.org/wiki/Iterator) and
the Decorator pattern (http://en.wikipedia.org/wiki/Decorator_pattern)

The goal as discussed in class is to adapt an existing iterator so that it returns filtered results.

As an example, f we placed the following text in a List with one word per entry, a normal iterator would return each word in order.
"Now is the time for all good men to come to the aid of their country."

But if we used a filtered iterator with a filterValue of "is" and "the" the result would be:
"Now time for all good men to come to aid of their country."

I have provided an abstract class for the filtered iterator, thus you need to supply FilteredIteratorImpl and provide a driver to demonstrate its effectiveness. The abstract class looks like:

```
import java.util.*;

public abstract class FilteredIterator<E1, E2> implements Iterator<E1>
{
	protected Iterator<E1> baseIterator;
	protected List<E2> filterValue;

	public FilteredIterator( Iterator<E1> baseIterator, List<E2> filterValue )
	{
		this.baseIterator = baseIterator;
		this.filterValue = filterValue;
	}

	public abstract boolean hasNext();

	public abstract E1 next();

	public abstract void remove();
}
```

I am also supplying a text file named melville.txt that contains some text you can filter.