**cs401 The Templated Containers**

**Tim Hoffman**

* [ArrayList](http://docs.oracle.com/javase/7/docs/api/java/util/ArrayList.html)
  + [ALDemo1.java](http://www.cs.pitt.edu/~hoffmant/401/TemplatedContainers/ALDemo1.java.txt)
  + [jumbles.txt](http://www.cs.pitt.edu/~hoffmant/401/TemplatedContainers/jumbles.txt)
* [HashMap](http://docs.oracle.com/javase/7/docs/api/java/util/HashMap.html)
  + [HMDemo1.java](http://www.cs.pitt.edu/~hoffmant/401/TemplatedContainers/HMDemo1.java.txt)
  + [StateCapitols.txt](http://www.cs.pitt.edu/~hoffmant/401/TemplatedContainers/StateCapitols.txt)
* [HashSet](http://docs.oracle.com/javase/7/docs/api/java/util/HashSet.html)
  + using Arrays.asList() [HSDemo1.java](http://www.cs.pitt.edu/~hoffmant/401/TemplatedContainers/HSDemo1.java.txt)
  + using loops [HSDemo2.java](http://www.cs.pitt.edu/~hoffmant/401/TemplatedContainers/HSDemo2.java.txt)
  + [sports.txt](http://www.cs.pitt.edu/~hoffmant/401/TemplatedContainers/sports.txt)
  + **OUTPUT**
  + **$ java HSDemo1**
  + **set1: [football, baseball] NOT a subset of set2: [hockey, track, lacrosse]**
  + **set1: [baseball, soccer] IS a subset of set2: [hockey, baseball, soccer]**
  + **set1: [football, hoops] IS a subset of set2: [football, baseketball, hoops]**
  + **set1: [soccer] IS a subset of set2: [soccer]**
  + **set1: [pole-vaulting, track, gymnastics] NOT a subset of set2: [drinking, girl-watching, partying]**
  + **set1: [golf, tennis] IS a subset of set2: [golf, tennis]**

**ITERATORS**

All the Collection containers have a built in iterator designed for efficient traversal of the object

**Wnehever you need to traverse a List sequentially you should use the built in iterator. Especially if your List isa LinkedList because if you use a for loop on a LinkedList every call to .get() has to *get* to that i'th element by starting back at the beginning of the list. Thus if you use anything but the iterator, a simple traversal of a Linked List in O(n^2) in stead of begin linear as it should. The iterator is also designed for removals during iteration. If your intend to remove elements as you traverse - use the iterator.**

Here is a sample program: [LinkedListIter.java](http://www.cs.pitt.edu/~hoffmant/401/TemplatedContainers/LinkedListIter.java) that should be used like this:

C:\> java LinkedListIter Strings.txt ints.txt

and here are the input files: [Strings.txt](http://www.cs.pitt.edu/~hoffmant/401/TemplatedContainers/Strings.txt) [ints.txt](http://www.cs.pitt.edu/~hoffmant/401/TemplatedContainers/ints.txt)

The program load the strings in an arraylist and then purges those strings that have an even length. It sdoes the same with a file of ints.

It is very inneficient to use any other mode besideshe iterator to traverse a LinkedList

The output for the program looks like this:

C:\Documents and Settings\hoffmant\Desktop>java LinkedListIter Strings.txt ints.txt

Original List

how

much

wood

would

a

wood

chuck

chuck

if

a

wood

chuck

would

chuck

wood

Purged List

how

would

a

chuck

a

chuck

would

chuck

Original List

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

Purged List

1

3

5

7

9

11

13

15