Java Collections Framework

Goal: Conciseness

A few concepts that are broadly useful
Not an exhaustive set of useful concepts

Two types of concepts are provided
Interfaces (i.e., ADTs)
Implementations

JCF Interfaces and Classes

<table>
<thead>
<tr>
<th>Interfaces</th>
<th>Classes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collection</td>
<td>HashSet</td>
</tr>
<tr>
<td>Set (no duplicates)</td>
<td>TreeSet</td>
</tr>
<tr>
<td>SortedSet</td>
<td>ArrayList</td>
</tr>
<tr>
<td>List (duplicates OK)</td>
<td>LinkedList</td>
</tr>
<tr>
<td>Map (i.e., Dictionary)</td>
<td>HashMap</td>
</tr>
<tr>
<td></td>
<td>TreeMap</td>
</tr>
<tr>
<td>Iterator</td>
<td></td>
</tr>
<tr>
<td>ListIterator</td>
<td></td>
</tr>
</tbody>
</table>

java.util.Collection (an interface)

public int size();
Return number of elements in collection

public boolean isEmpty();
Return true if collection holds no elements

public boolean add (Object x);
Make sure the collection includes x; returns true if collection has changed (some collections allow duplicates, some don’t)

public boolean contains (Object x);
Returns true if collection contains x (uses equals() method)

public boolean remove (Object x);
Removes a single instance of x from the collection; returns true if collection has changed

public Iterator iterator ();
Returns an Iterator that steps through elements of collection

java.util.Iterator (an interface)

public boolean hasNext ();
Returns true if the iteration has more elements

public Object next ();
Returns the next element in the iteration; throws NoSuchElementException if no next element

public void remove ();
The element most-recently returned by next() is removed from the collection; can throw IllegalStateException if next() not yet used or if remove() already called

Additional Methods of Collection

public Object [] toArray ();
Returns a new array containing all the elements of this collection

public Object [] toArray (Object [] dest);
Returns an array containing all the elements of this collection; uses dest as that array if it can

Bulk Operations:
public boolean containsAll (Collection c);
public boolean addAll (Collection c);
public boolean removeAll (Collection c);
public boolean retainAll (Collection c);
public void clear ();
java.util.Set (an interface)

- Set extends Collection
- Set has no methods of its own, but it inherits the methods from Collection
- A Set contains no duplicates
- If you attempt to add() an element twice then the second add() will return false (i.e., the Set has not changed)
- Write a method that checks if a given word is within a Set of words
- Write a method that removes all words longer than 5 letters from a Set
- Write methods for the union and intersection of two Sets

Set Implementations

- java.util.HashSet (a hashtable)
  - Constructors
    - public HashSet ();
    - public HashSet (Collection c);
    - public HashSet (int initialCapacity);
    - public HashSet (int initialCapacity, float loadFactor);
- java.util.TreeSet (a balanced BST [red-black tree])
  - Constructors
    - public TreeSet ();
    - public TreeSet (Collection c);
...