

## Java Collections

- Java has a rich hierarchy of classes for collections of various kinds.
- Some important interfaces:
  - Collection (at the root of the hierarchy).
  - Set
    - SortedSet
  - List
  
  - Map (for dictionaries – not a collection)
    - SortedMap

CSE 341, Winter 2003

1

## The Collection Interface

- Some important methods in Collection:
  - int size()
  - boolean isEmpty()
  - boolean contains (Object element)
  - Iterator iterator()
  - Object[] toArray()

CSE 341, Winter 2003

2

## The Map Interface

- Some important methods in Map:
  - int size()
  - boolean isEmpty()
  - boolean containsKey (Object key)
  - boolean containsValue (Object value)
  - Object get (Object key) // look up operation
  - Object put (Object key, Object value)
  - Set keySet()
  - Collection values ()

CSE 341, Winter 2003

3

## Some Important Collection Classes

- HashSet (implements Set)
- ArrayList (list implemented using a resizable array)
- LinkedList
  
- HashMap (implements Map)
  
- Obsolete (deprecated): Vector, Hashtable (alas the book uses these)

CSE 341, Winter 2003

4

## The Iterator Interface

- Principal methods
  - boolean hasNext()  
*are there more elements? (no side effects)*
  
  - Object next()  
*get next element (side effect – advance internal pointer)*
- Obsolete (deprecated): Enumeration (alas the book uses this also)

CSE 341, Winter 2003

5

## Iterator Example

```
// example of use – print each element in an
// ArrayList alist

Iterator i = alist.iterator(); ...
while (i.hasNext()) {
    System.out.println(i.next());
}
```

CSE 341, Winter 2003

6

## Defining Iterators

```
class MyArray {
  Object[] a;
  public MyArray (int size) {...} // constructor
  public Object at (int i) {...} // access an element
  public void set (int i, Object value) {...}

  class MyIterator implements Iterator {
    // an inner class! ... }
    public Iterator iterator () {...}
    ...
  }
}
.class files produced by compiler:
MyArray.class, MyArray$MyIterator.class
```