**Arrays and ArrayList**

Bjarke Hammersholt Roune

October 10, 2013

**The difference between arrays and ArrayLists**

A Java array is pretty much the same as a Java ArrayList, so if you understand one you understand the other, you just have to learn a new syntax to use arrays. They do the same thing except that arrays cannot be made larger after construction, so you have to start by saying how many elements you want the array to have.

**An example**

Suppose we want to construct an ArrayList of Strings, add two Strings to it and then print those strings out twice using the two different kinds of for-loops. We show how to do this using both ArrayLists and arrays.

**ArrayList Array**

ArrayList<String> strs; String[] strs;

strs = new ArrayList<String>(); strs = new String[2];

strs.add(“Hello”); strs[0] = “Hello”;

strs.add(“world”); strs[1] = “world”;

for (int i = 0; i < strs.size(); i++) for (int i = 0; i < strs.length; i++)

System.out.println(strs.get(i)); System.out.println(strs[i]);

for (String str : strs) for (String str : strs)

System.out.println(str); System.out.println(str);

**A dictionary**

To be more precise, this is how to do things with an ArrayList and an array for containing some type T. Here T could be String, as in the example above, or T could be int or double or whatever kind of object you wish to have a container for. In the code below i stands for some specific integer, like 5, and e is a reference to some object of type T.

**Thing to do ArrayList Array**

Declare variable ArrayList<T> a; T[] a;

Construct object a = new ArrayList<T>(); a = new T[i];

Set element at index i a.set(i, e); a[i] = e;

Append element a.add(e); Not possible!

Get element at index i a.get(i) a[i]

get the length a.size() a.length

**Why are there arrays in Java?**

We might ask why there even are arrays, if ArrayList does everything that an array does. The answer is that arrays are slightly more efficient than ArrayLists, which is to say that programs written using arrays can sometimes do their computations in less time than if they were written using ArrayLists. ArrayLists also require more memory (RAM) than arrays do. The fact that arrays cannot grow is what allows them to be implemented more efficiently.