

Project 6 – 20 Points

Write a Java program to demonstrate the use of linked-lists. You will **not** be using the `LinkedList` from `java.util`. Rather, you will be creating your own `OrderedList` class and implement a linked list with reference variables. In addition you will implement the `Iterable` and `Iterator` interfaces. This exercise is intended to build the tools necessary to understanding complex memory management (although not to the degree the C programming language grants) rather than using an array-backed class like `Vector`. In addition, your class will have the `Iterable` capability that will allow you to use your class objects in a for-each loop.

The key to this design is that the list will be ordered (sorted ascending) and will not contain duplicates. Your code will maintain this order.

Refer to the class notes on the visual construction of a linked list.

You will work with three file names provided (in the order below) on the command line:

```
d6.dat
d6.delete
d6.out [or whatever you call it on the command line]
```

The files `d6.dat` and `d6.delete` will be provided for you. You will produce an *outputfile* that contains all of `d6.dat` in sorted order, minus the duplicates and minus any items specified in `d6.delete`.

The solution is as follows:

Create an `OrderedList` class that implements the `Iterable` interface. This will define the inner classes `OLIterator` and `Node`.

The `Node` class simply defines instance variables for the `data` and the reference to the `next` node in the list in addition to a simple constructor. The `OLIterator` class will implement the `Iterator` interface which includes methods for `hasNext()`, `next()` and `remove()`. Your `remove()` method need not do anything but must be present to satisfy the interface.

The final pieces of the `OrderedList` class are the instance variable for the `head` of the list, a constructor and the methods:

```
boolean isEmpty();
boolean insert(String item);
void delete(String item);
```

Once you have processed the `d6.dat` file and the `d6.delete` file, it is a simple matter of iterating over the list and writing the object data to the *outputfile*.

The output file size will be 22,725 or 27,270 bytes depending on platform and methods used.

You should submit your project components as:

```
$ submit jojo d6.java  
$ submit jojo OrderedList.java
```