6.005 Lecture 10: Data Abstraction in Java

Lecture Exercise

**Due at start of next lecture** (Fri Oct 12)

This exercise gives you more practice with abstract data types. For each of the following ADT representations, write a rep invariant and an abstraction function (as mathematical statements, not as Java code) and illustrate each one with a picture showing some legal rep values, some illegal rep values, and the abstract values that the legal reps map to.

1. **IntSet** represents a set of integers as an array. Example of an abstract value: {5, 200, 393, -2}

   ```java
   public class IntSet {
       private int[] array;
       ...
   }
   ``

2. **DigitSet** represents a subset of the digits 0-9 using a single integer. Example of an abstract value: {0, 3, 8}

   ```java
   public class DigitSet {
       private int n;
       ...
   }
   ``

3. **Uncertain** represents a number with a degree of uncertainty. Example of an abstract value is 0.6 ± 0.1. The desired representation should use the low end and high end of the range (in this case, 0.5 and 0.7), rather than the mean and error spread.

   ```java
   public class Uncertain {
       private double low;
       private double high;
       ...
   }
   ``

4. **Chromosome** represents a double helix of DNA as a pair of strings, one for each strand of DNA. (See http://ghr.nlm.nih.gov/handbook/basics/dna if you don’t recall the basic structural constraints of DNA.)

   ```java
   public class Chromosome {
       private String r;
       private String s;
       ...
   }
   ```