**Instructions**

Imagine that you have received a shipment containing a large number of glass jars and a separate shipment of the lids for all of those jars.  There is a matching lid for each jar and there are exactly the right number of each.  Your task is to find the lid for each jar. When trying to place the lid on a jar, you can see whether it is too small, too large, or fits.  You cannot compare two lids directly and cannot compare two jars directly.   
   
Your assignment is to develop and implement an efficient algorithm to solve this problem.  You must use the JarsAndsLids.java driver and the classes found in Lid.java and Jar. Java.  You **may not** use the getSizeMethods().  You may only call the fit() methods.  You may not modify the Jar and Lid classes.  You’ll need to write a static method called *arrange*.  Place that static method in a class called *Arrange.* You cannot create additional arrays and must only use  constant extra space.  
  
Here is a sample output.  The first number in each pair is the original location of the item in the array. The second number is the size.  (In the example below, the lid in original position 5 fits the jar that was originally in position 0.  After the jars and lids are arranged, the jar and lid size 13 are found in position 0 in their respective arrays.

Can sort them  
  
Original List  
Lids:                          [0.84, 1.20, 2.71, 3.85, 4.26, 5.13, 6.78, 7.88]  
Jars:                           [0.13, 1.88, 2.71, 3.78, 4.20, 5.85, 6.26, 7.84]  
   
Lids and Jars Match: true  
   
Matching Lids and Jars  
Lids:                          [5.13, 1.20, 4.26, 2.71, 6.78, 0.84, 3.85, 7.88]  
Jars:                           [0.13, 4.20, 6.26, 2.71, 3.78, 7.84, 5.85, 1.88]