WordUtilities.java

Objective: To implement binary searching on a word database.

Background:
Binary search is a very efficient way to find a specific thing in a large database. In this project, you will be given a large database of words, in random order, and asked to determine if a specific word is in the database.

Binary search requires the data to be in sorted order, either ascending or descending, You will create a modified version of merge sort from the last project to first arrange the words in ascending order.

Assignment:

1. Download the **WordUtilities.zip** file from Mr Greenstein's web site and unzip. It will create a **WordUtilities** directory. Do all of your work in the directory. You are provided three files in the directory, specifically **WordUtilities.java**, **SortMethods.java**, and **randomWords.txt**. You need to copy **FileUtils.java** to the **WordUtilities** directory so it can open and read the **randomWords.txt** file.

2. **SortMethods.java** has an incomplete **mergeSort** method. Complete the **mergeSort** method to handle an **ArrayList** of **String**s. You may use the **mergeSort** method from your previous sorting project as a model.

3. **WordUtilities.java** contains the binary search method. You will complete both a recursive version (**binarySearchRecurse**) and an iterative version (**binarySearchIterative**) of the search process. The **WordUtilities** class contains a main method for testing your program. A sample run is shown below.

Here is a sample run output of **WordUtilities**:

```
% java WordUtilities

Testing findWord method
-----------------------
"hello" found, index = 20038
"foo" NOT found
"utilitarian" found, index = 45727
"frufru" NOT found
"student" found, index = 41276
"fubsy" NOT found
"pulchritude" NOT found
"callipygian" NOT found
"whithersoever" found, index = 47871
```