CS 251 Intermediate Programming
Lab 6: Sorting Lines in a File

Brooke Chenoweth

Fall 2018

This assignment should help you get a bit more familiar with file I/O classes. It also gives you another chance to work with collections and to remember how to handle command line arguments.

Problem Description

You will write a program that will take the names of two text files as command line arguments. (Only the first file needs to exist. You'll be writing the second one.) It will read all the lines from the first file (ignoring some), sort them by length, and write them to the second file. Any line that begins with a '#' (pound sign) character should be ignored.

The lines should be sorted so that the shortest line is first. (Note: comparing length is not the default ordering for Strings.) If two lines are the same length, compare them in lexicographically to break the tie, and sort them in lexicographical order. (The natural ordering of the String type compares lexicographically, so "abc" will come before "def".)

What do you need to do?

Write one class named `LineSorter` that will solve the problem described above. This is a small enough task that you will likely be able to most of it in your `main` method.

Don't overthink this assignment. Using existing libraries will make your program very simple. My solution has less than 40 lines of code, for example.

Example

If you had a file named `testin.txt` that contained the following:

```
These are some test lines
short line
abcdefghij
this line is much longer than the short line
123 456
```

```
#Ignore this line
Blah blah blah
# ignore this, too
```

After you ran the command `java LineSorter testin.txt testout.txt`, you would find that `testout.txt` contains the following:

```
123 456
abcdefghij
short line
Blah blah blah
These are some test lines
this line is much longer than the short line
```

Turning in your assignment

Once you are done with your assignment, use UNM Learn to turn in `LineSorter.java`.