Assignment #3

```
Original Due: 3:00 PM, Monday, March 22
    Extended: 3:00 PM, Monday, March 29
```

You must complete this assignment by yourself. You cannot work with anyone else in the class or with someone outside of the class. You are not allowed to copy solutions from the world wide web. The code you write must be your own.

Provided Files:

- A3.java - A shell file with the main method.
- input.txt - An example (large) data file.
- output.txt - The required output for the example input.txt file.
- sample1.txt - A small data file for testing.
- sample2.txt - A small data file for testing.
- sample3.txt - A small data file for testing.
- sample4.txt - A small data file for testing.
- sample5.txt - A small data file for testing.
- sample6.txt - A small data file for testing.

Description: Write a program that creates a linked list that has `Bank`-type data. Below is the definition of class `Bank`:

```
class Bank {
     public String bankA;
     public int year;
     public double sizeofA;
     public String bankB;
     public double sizeofB;
     public double diff;
}
```

Your program shall repeatedly (i) read one line at a time from input.txt, (ii) create the corresponding node with a `Bank`-type data instance, and (iii) add it to the linked list in the correct place so that the nodes have `year` values in ascending order. The new node has to be the last node within each `year`. You are NOT allowed to read the entire data entries, sort them all, and then create the linked list at once.

You must use the shell file for this assignment. You are not allowed to remove or change existing code in the shell file. You CANNOT use Java class `java.util.LinkedList` for this assignment. The shell file contains a single placeholder that indicates where you fill in your code.

Read the next page.

Below are a few input samples and the answer output:

Sample (1):

bankA	year	sizeofA	bankB	sizeofB	diff
1	2001	0.2	2	0.4	0.2
3	2001	0.1	5	0.4	0.3
2	2000	0.4	4	0.9	0.5
1	2000	0.2	3	0.1	0.1

Sample (2):

bankA	year	sizeofA	bankB	sizeofB	diff
1	2001	0.2	2	0.4	0.2
2	2000	0.4	4	0.9	0.5
3	2001	0.1	5	0.4	0.3
1	2000	0.2	3	0.1	0.1

Sample (3):

bankA	year	sizeofA	bankB	sizeofB	diff
2	2000	0.4	4	0.9	0.5
1	2001	0.2	2	0.4	0.2
3	2001	0.1	5	0.4	0.3
1	2000	0.2	3	0.1	0.1

Required output:

bankA	year	sizeofA	bankB	sizeofB	diff
2	2000	0.4	4	0.9	0.5
1	2000	0.2	3	0.1	0.1
1	2001	0.2	2	0.4	0.2
3	2001	0.1	5	0.4	0.3

Read the next page.

Sample (4):

bankA	year	sizeofA	bankB	sizeofB	diff
2	2000	0.4	4	0.9	0.5
1	2000	0.2	3	0.1	0.1
1	2001	0.2	2	0.4	0.2
3	2001	0.1	5	0.4	0.3

Sample (5):

bankA	year	sizeofA	bankB	sizeofB	diff
1	2001	0.2	2	0.4	0.2
2	2000	0.4	4	0.9	0.5
1	2000	0.2	3	0.1	0.1
3	2001	0.1	5	0.4	0.3

Sample (6):

bankA	year	sizeofA	bankB	sizeofB	diff
2	2000	0.4	4	0.9	0.5
1	2001	0.2	2	0.4	0.2
1	2000	0.2	3	0.1	0.1
3	2001	0.1	5	0.4	0.3

Required output:

bankA	year	sizeofA	bankB	sizeofB	diff
2	2000	0.4	4	0.9	0.5
1	2000	0.2	3	0.1	0.1
1	2001	0.2	2	0.4	0.2
3	2001	0.1	5	0.4	0.3

Read the next page.

Given the input data in input.txt, your program should produce the same output in output.txt through console (not a file). Even minor differences in output will cause you to lose points. Use a diff tool to ensure your program produces the correct output.

You may want to use the following commands that divert the standard output to a file and then compare files:

Windows users:

```
javac A3.java
java A3 input.txt > my.txt
fc my.txt output.txt
```

MacOS users:

```
javac A3.java
java A3 input.txt > my.txt
diff my.txt output.txt
```

Another suggestion is to use a text editor tool like Notepad++ that has a built-in diff tool:

https://www.thewindowsclub.com/how-to-compare-two-files-in-notepad

Submission: your **A3.java** file.

General Programming Assignment Requirements:

- Classes must be in the default (no package statement) unless otherwise specified. You will lose 30 points (all points) if you put a package statement in your program.
- If your program that does not compile or does not run, you will lose all points.
- If you submit the wrong file your grade will suffer accordingly. Most likely a 0.
- You must fill in the header for every file you submit or you may lose points.

Checklist: Did you remember to:

- worked on the programming assignment by yourself?
- fill in the header in your file **A3.java**?
- ensure your program does not suffer a compile error or runtime error?
- properly indent your source code so that your indenting is readable and consistent?
- use good names for variables to make your program easy to understand?
- turn in your Java source code in a file named **A3.java** through D2L?