This work complies with the JMU Honor Code. I have neither given nor received unauthorized assistance, and I will not discuss the exam contents with anyone who has not taken it for credit.

Name:_____   Signature:_____

Tweet Tester

Twitter is an online social networking service that enables users to send and read short 140-character messages called tweets. You have been chosen to help make a custom Twitter app for JMU students.

For this exam, you need to create two classes from scratch: **Tweet** and **TweetTest**. The first class must implement the **stats** method described below. The second class should use JUnit to test your code thoroughly. At the end of (and during) the exam, you will submit these two files via Web-CAT. You may submit as many times as you like; only the last submission will be graded.

The `stats` method takes a single string parameter that represents a tweet. It returns a string in the format `"%d-%d-%d"` representing the number of mentions, hashtags, and links in the given tweet.

- Every mention will start with the `'@'` character and have at least one non-space or non-tab character following it (e.g., @mayfiecs).
- Every hashtag will start with the `'#'` character and have at least one non-space or non-tab character following it (e.g., #cs149).
- All links will start with the string `"http://"`. Twitter ignores case, so HTTP and http are counted the same. You do not need to check for any characters following `"http://"`.

You must write complete documentation comments for both classes (including your name and today's date) and all methods (including @param and @return tags). As with previous assignments, the code you submit must pass Checkstyle without any errors. It's strongly recommended that you complete the exam in this order:

1. Write documentation and stubs for all classes and methods. [10 min]
2. Implement your test cases; they should all fail at this point. [10 min]
3. Submit to Web-CAT to make sure that it compiles and runs. [5 min]
4. Implement/test mentions and hashtags; submit to Web-CAT. [15 min]
5. Implement/test http links ignoring case; submit to Web-CAT. [10 min]

You may use the back of this sheet for scratch paper. Good luck!