Assignment 3: Buffon's Needle

Dr. William Kreahling

March 30, 2020

1 Overview

For this assignment you will write a multi-threaded application that estimates the value of PI by Dropping needles on a floor marked with evenly spaced lines. Each thread will calculate multiple "drops" of a needle.

One way to calculate your estimate is: $\frac{2N(hits + misses)}{L(hits)}$

2 Instructions

1. Class Buffon will print the correct answer AFTER all threads have finished.
2. Class Buffon will prompt the user for the number of experiments, the number of threads, the length of the needle(N) and the distance between lines (L) (for this experiment the size of the needle should be less than the distance between lines).
3. All threads must be started and run concurrently.
4. Class Buffon, should start accessing return values from threads as soon as they become available. (e.g., if thread 7 is done before thread 100, access thread 7 as soon as it is finished)
5. Your Experiment class accepts: a unique id, the distance between lines, the length of the needle and the number of experiments to conduct.
6. You must use message passing to transport data, you may **not** use shared memory.
7. Style and code correctness count.
8. No methods larger than 40 lines
9. No static methods or variables (except main), without good reason.
10. All exceptions should be handled in an appropriate manner.
11. Correct input is not guaranteed – you should print useful error messages!
12. Helper methods should have correct access modifiers.
13. You must use Java for this assignment.

3 Examples:

```
I will demonstrate a simplified version via video.
```

4 Notes on Collaboration

You may work in teams of up to two on this assignment. Note that all members of a team will receive the same grade on the assignment.

5 Hand-In Instructions

This assignment is due by 11:59 PM on Monday April 13th. A single version version of the assignment is due from each team. Submit all source files associated with the program as well as the Makefile. To submit your files, use the *handin* command on agora. Handin works as follows:

```
handin.<course#>.<section#>  <assignment#>  <files>
```

Therefore, to submit this assignment, you must use the following command (assuming each of the files is in your current working directory):

```
handin.370.1  3  *.java
```