

# Lab 07 Design

Scott Rizzo

November 27, 2018

## 1 Class Diagram

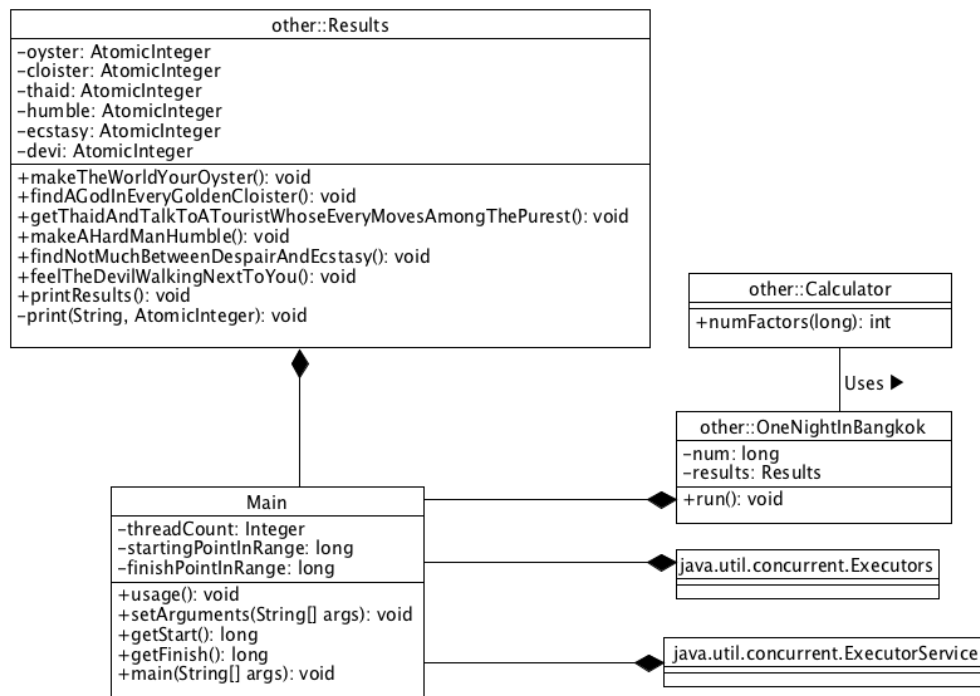


Figure 1: Class Diagram

## 2 Class Responsibilities

### 2.1 Calculator

The `Calculator` is responsible for calculating the number of factors of a given number.

### 2.2 Result

`Result` is responsible for incrementing specific `AtomicInteger` variables whenever certain methods are invoked.

### 2.3 OneNightInBangkok

The `OneNightInBangkok` is responsible for invoking certain methods in the `Result` class based on number of factors returned by `Calculator` class of a number.

## 2.4 Main

The **Main** class is responsible for taking command line arguments and based on those arguments run the program. In main for this lab was able to stumble upon something called Executors class, and ExecutorService interface which extends Executor interface. Using a method of Executors I was able to create a thread pool with the threads given in command line arguments. While looping through the appropriate amount of values based on what was given for the range in arguments I invoke the execute method giving it the OneNightInBangkok object to add to the queue in the thread pool. By the invoking another method which says while tasks(OneNightInBangkok's) are not done keep waiting, and when they are done shutdown and print results.