CSC565: Operating Systems

Spring 2017

Assignment 3

Due Date: April 13, 2017 -- Due Time: 23:59

Project: Process Synchronization using C/Linux and Java [100 points]

Implement the following project using:

a) C/Linux, and
b) Java

A *barrier* is a tool for synchronizing the activity of a number of threads. When a thread reaches a *barrier point*, it cannot proceed until all other threads have reached this point as well. When the last thread reaches the barrier point, all threads are released and can resume concurrent execution.

Assume that the barrier is initialized to N – the number of threads that must wait at the barrier point:

```
init(N);
```

Each thread then performs some work until it reaches the barrier point:

```
/* do some work for awhile */

Barrier_point();

/* do some work for awhile */
```

Using some of the synchronization tools discussed in class, construct a barrier that implements the following interface (API):

- int init(int n) – Initializes the barrier to the specified size.
- int barrier_point(void) – Identifies the barrier point. All threads are released from the barrier when the last thread reaches this point.

The return value of each function is used to identify error conditions. Each function will return 0 under normal operation and will return -1 if an error occurs.

Also provided are a file in C and a file in Java that you must use to test your programs.