

# CS 6378-001: Project I

Instructor: Ravi Prakash

Assigned on: September 21, 2002

Due date and time: October 27, 2002, 11:59 pm

Implement Raymond's token-based algorithm for distributed mutual exclusion as described in Section 6.13 of the textbook. This is an individual project and you are expected to demonstrate its operation to the instructor and/or the TA.

## 1 Requirements

1. There are fifteen nodes in the system, numbered from 1 to 15. Initially, the tree-structure reflected by the *holder* variables in the nodes represents a balanced binary tree. Node 1 is the root node and holds the token. Nodes  $2i$  and  $2i + 1$  are the two descendants of node  $i$ .
2. There are reliable socket connections (TCP) between each pair of nodes related by the *holder* relationship. The algorithm's messages are sent over these connections.
3. Each node goes through the following sequence of operations until each node has successfully exited the critical section 20 times:
  - (a) While not in the critical section, waits for a period of time that is uniformly distributed in the range [5, 10] time units before trying to enter the critical section.
  - (b) On entry into the critical section, exits it after 3 time units.
4. Subsequently, nodes numbered 2, 4, 5, 8, 9, 10, and 11 continue to issue requests for entry into the critical section at the same rate as before. Nodes numbered 1, 3, 6, 7, 12, 13, 14 and 15, after exiting the critical section, wait for a period of time that is uniformly distributed in the range [45, 50] time units before trying to enter the critical section. As earlier, the critical section execution lasts 3 time units.
5. Once a node has successfully exited the critical section 40 times (including the 20 mentioned earlier), it does not make any more attempt to enter the critical section, and sends a *completion notification* to node 1.
6. Node 1 brings the entire distributed computation to an end once its has received *completion notification* from all the nodes, including itself.

## 2 Data Collection

For your implementation of the mutual exclusion algorithm report the following:

1. The total number of messages exchanged.
2. The total number of REQUEST messages sent by each node (including the REQUEST messages generated by it on receiving a REQUEST message from a neighbor).
3. The maximum and minimum length of the *request\_q* at each node during the execution.
4. For each node, report the following for each of its attempts to enter the critical section: the elapsed time between making a request and being able to enter the critical section.

## 3 Point Distribution

**Implementation (50%):** Source code of your well structured and well documented program. You may write your code in C++ or Java.

**Correctness (50%):** Output that your program produces and the statistical analysis of the results.

## 4 Submission Information

The submission should be through WebCT in the form of an archive consisting of:

1. File(s) containing the source code.
2. The `makefile`.
3. The output file.
4. A short report (text file) containing the analysis of the results in the output file, and information about the platform/machines on which you executed your program.

Please do not submit the executable file. Also, any code that is not written by you, but is obtained from some other source should be properly acknowledged in the source file, as well as in the report.