

Grading policy for first programming assignment

10pts: Lamport's algorithm: You should present evidence that you implemented logical clocks. For that you should show that the computers update their logical counter each time they receive a message. Also, slower computers should correct their counters when they receive messages if they are lagging behind.

10pts: Print statements (When trying to use a shared resource): You should make each computer print if it:

- Plans to use the shared resource
- Receives OK from the other computers
- Acquires the right to use the shared resource
- Releases the shared resource

10pts: Print statements (When receiving requests to use a shared resource): You should make each computer print if it:

- Receives a request from another computer to use a shared resource while not planning to use it.
- Receives a request from another computer to use a shared resource while using it. The computer should queue the request and send an OK after releasing the resource.

10pts: Ricart-Agrawala algorithm: You should show that algorithm actually protects the shared resource. If one computer acquires the right to use the shared resource, other computers should not be allowed to access it.

10pts: Your written report: It should not be very extensive, but you should clearly explain how did you implement the Ricart-Agrawala algorithm, e.g., with/without threads, how do you queue incoming requests, did you use mutexes and why or why not.

NOTE: Our recommendation is for you to emulate as if the three computers run at a different clock speeds and randomly trying to use the shared resource every now and then. In that way, by simply looking at the print statements for several seconds most (if not all) of the cases explained above should occur.

A sample output is attached in the next page for your reference.

