Eric Blasko

Tong Yu

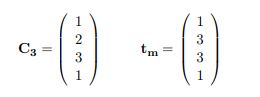
CSE 461

March 9, 2019

\*\*Note: extra credit questions were completed. Since the questions did not state how much they are worth, I am assuming the base score is 50 with each extra credit question being 10 each.

Homework # 4

1. **Suppose the “Birman-Schiper-Stephenson Protocol” is used to enforce “Causal Ordering of Messages” of a system that has four processes, P1, P2, P3, and P4. With the help of diagrams, explain clearly what the process would do in each of the following cases.**
2. **The current vector time of Process P3 was C3 when it received a message from P2 along with vector time stamp tm, where**

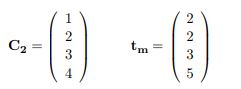
****

**Should P3 deliver the message immediately? Why? If not, what should it do?**

Answer:

Yes, it would deliver. This is because all of C3[i] has been received from other Pi prior to P3 received a new message from P2 along with its timestamp.

1. **Process P2 with current vector time C2 received a message from P1 along with vector time stamp tm, where**

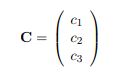
****

**Should P2 deliver the message immediately? Why? If not, what should it do?**

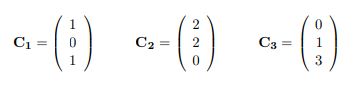
Answer:

No, it cannot be delivered. This is because the P2 has not received any message from P4. Thus, P2 must buffer the message till it receives an update from P4.

1. **Consider a cut C:**

****

**where c1, c2, and c3 are the cut events with vector clocks C1, C2, C3 respectively:**

****

**Calculate TC = sup(C1, C2, C3). Is C a consistent cut? Why?**

Answer:

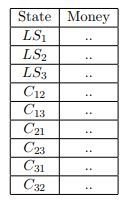
Tc = , Ci[i] does not equal Tc[i]

This is an inconsistent cut because it violates the theorem which states that

Tc = for the cut to be consistent.

1. **A banking system uses Chandy-Lamport global state recording protocol (Snapshot Algorithm) to record its global state; markers are sent along channels where FIFO is assumed. The system has three branches P1, P2, and P3 and are connected by communication channels Cij , where i, j = 1, 2, 3. Suppose LSi denote the local state ( the money the branch possesses at the time of recording ) of branch Pi .**

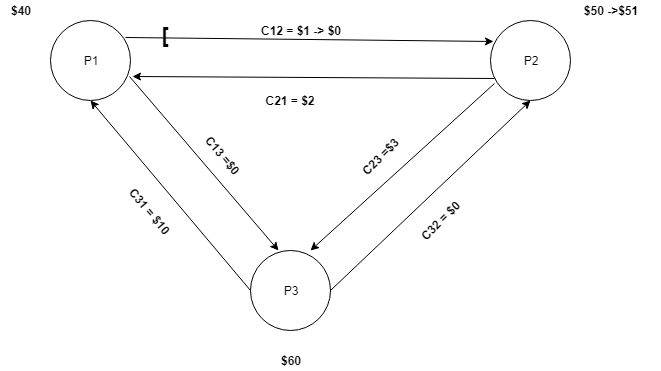
**P1 initiated the recording process. Right before P1 sent out the marker, a $1 transaction was in transit on C12, a $2 transit on C21, a $3 transit on C23, and a $10 transit on C31 ( assume 1 that the units are in million dollars ) and branches P1, P2, and P3 had $40, $50, and $60 respectively ( not including any money in transit ). Assume that the branches do not send out any other money during the whole recording process and the markers from P1 arrived at other banks earlier than other markers. With the help of diagrams, find out the state LSi of Pi and channel states Cij where i, j = 1, 2, 3. Tabulate your results in the following format:**

****

**Show your steps clearly**

Answer:

|  |  |
| --- | --- |
| State | Money |
| LS1 | $40 |
| LS2 | $51 |
| LS3 | $60 |
| C12 | $0 |
| C13 | $0 |
| C21 | $2 |
| C23 | $3 |
| C31 | $10 |
| C32 | $0 |



1. **In Lamport’s algorithm for mutual exclusion, Process Pi enters CS when the following 2 conditions are satisfied:**
2. **Pi ’s request is at the head of requestiqueuei**
3. **Pi has received a ( REPLY ) message from every other process time-stamped later than tsi**

**Condition 1) can hold concurrently at several sites. Why then is 1) needed to guarantee mutual exclusion?**

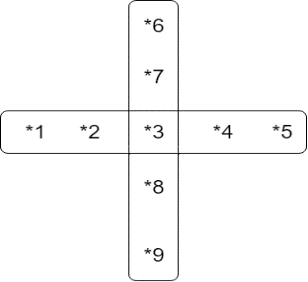
Answer:

Each process must obtain an internal queue if there is only one process able to enter the critical section. It would allow a process to compare the timestamp of its own request along with other timestamp from other processes. Consequently, the lack of a queue at each process, will make the second condition of requiring a reply from every other process not possible.

**Does the algorithm work if condition 2) is removed? Why? Give an example with illustrations ( drawings ) to support your argument.**

Answer:

According to Mackawa’s voting algorithm that the process needs to occur only in a group of simple majorities that shares some of its members with another groups. For shared processes to send replies to requesting processes of other groups, the active process must release the vote of the shared members.



1. **In Lamport‘s algorithm of mutual exclusion, if a site Si is executing the critical section, is it necessary that Si ‘s request need to be always at the top of the request-queue at another site Sj? Explain and give an example ( with diagrams ) to support your argument.**

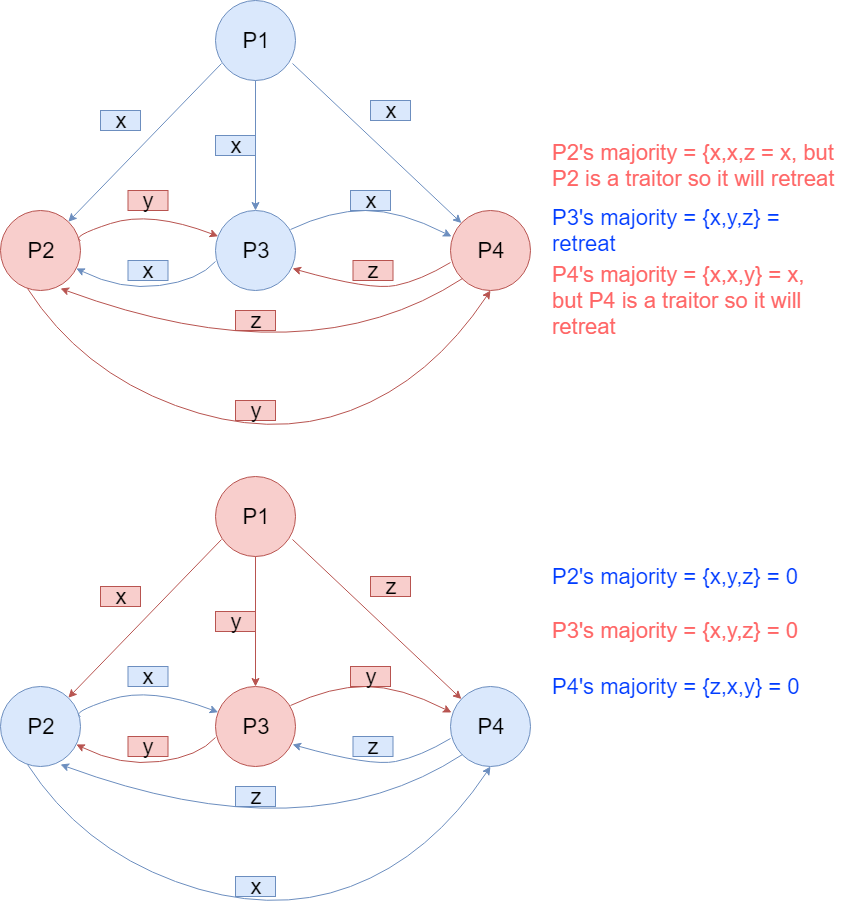
Answer:

No, it’s not necessary, because if it is executing the critical section, that means it has already been to the critical section, so it does not matter if it’s on the top or not. When exiting the critical section, process Pi removes its request from the head of its request-queue and sends a timestamped release to every other process

1. **Can Byzantine agreement be always reached among four processors if two processors are faulty? With the help of diagrams, explain your answer.**

Answer:

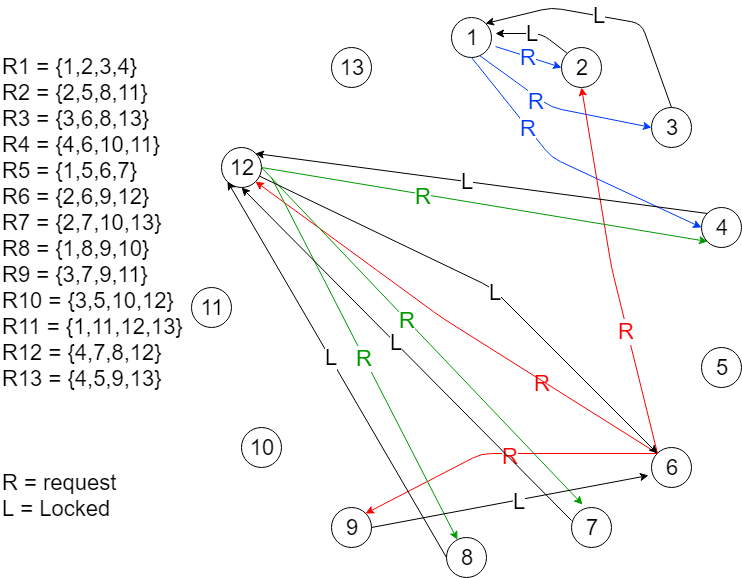
According to lamport-shostack-Pease algorithm the agreement cannot be reached if the number of faulty processors is 1/3 of the total number of processors. As a solution, a Byzantine agreement can only be reached among four processors iff there is less than one third faulty processors and here the maximum of faulty process should be one.



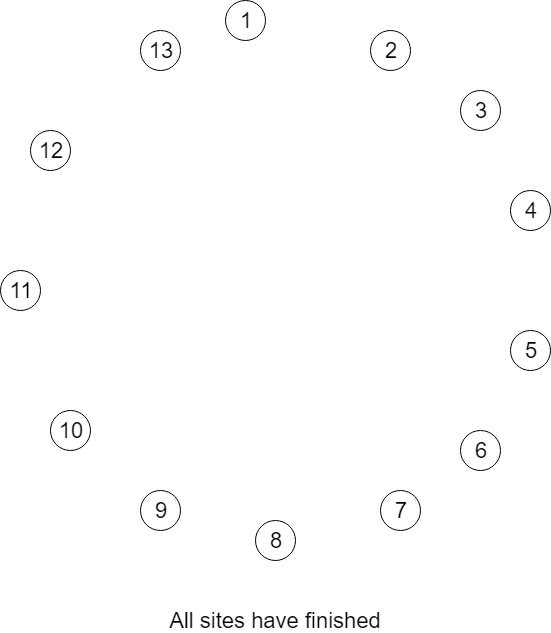
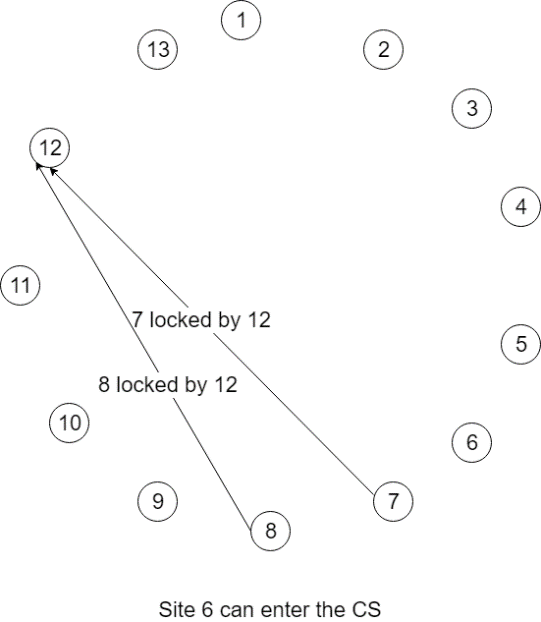
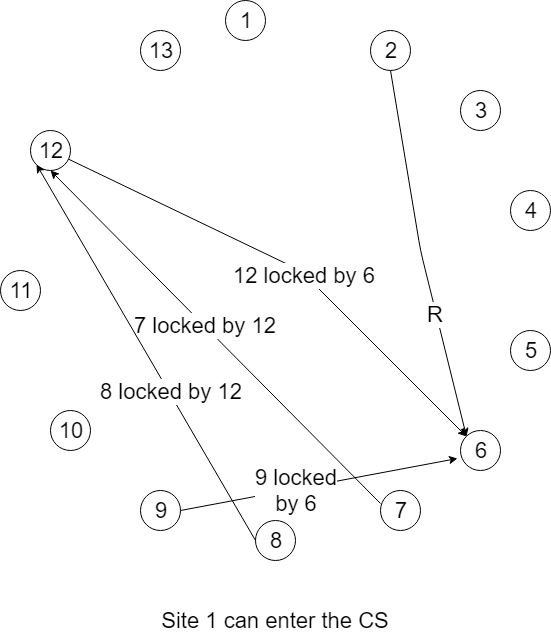
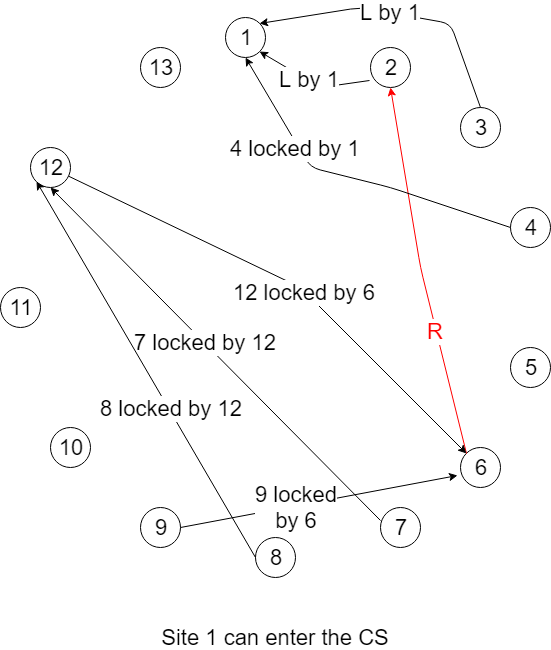
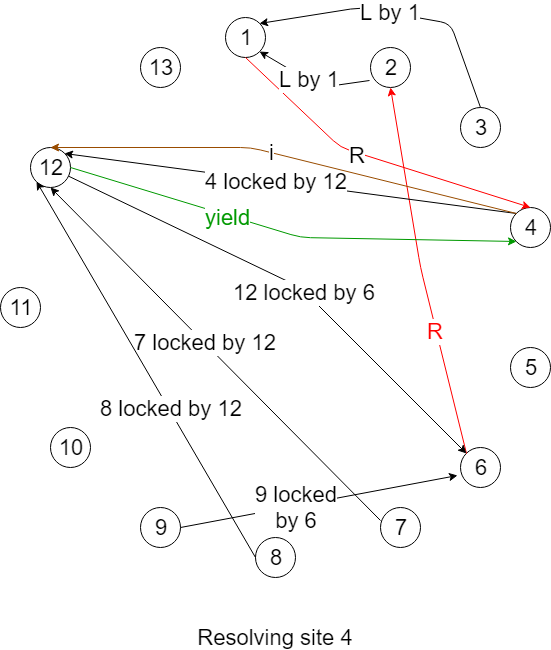
1. **Maekawa’s Algorithm is used to achieve mutual exclusion for 13 sites. Suppose the sites are labeled 1, 2, ..., 13. Find the request sets R1, R2, ... , R13. Suppose sites 1, 6, 12 want to enter a critical section ( CS ) and they have sent requests in the order 1, 6, 12. The following sequence of events have occurred in the order listed:**
2. **The requests of site 1 have arrived at site 2, and site 3. Its request to site 4 is on the way.**
3. **The requests of site 6 have arrived at site 9, and site 12. Its request to site 2 is on the way.**
4. **The requests of site 12 have arrived at site 4, and site 7 and 8.**

**Draw a diagram to show which sites have been locked and locked by whom.**

**Suppose the transit requests have arrived at their destinations. At this point can any site enter the CS? Why? If not, how do the sites resolve the problem?**

****

None of the sites can enter the critical section because they must resolve the deadlock first. In order to resolve the deadlock site# 4 should send inquire message to site #12. But, site #12 have been already locked by site #6; therefore, the inquire message from site #4 must be forward to site# 6. However, here the inquiry message would find the site #6 is lower priority than site #1 that have a request in wait to lock site #4. Thus, site #4 supposed to release itself from site #12 by letting site #12 reply to the inquire message by a yield message. Once site #4 receives the yield message, it become available for site #1 to acquire. Here, site #1 became able to enter the critical section. At the time site #1 finish the critical section, it will release all the locked sites #2,3,4. Site #6 now can enter the critical section, and after finish site #12 will also be able to enter the critical section.

****

1. **In a distribution system, there are 20 servers. Suppose the utilization of a server is 60%. Calculate the probability that the system has at least one task waiting and at least one server lying idle. Show your steps clearly**

Answer:

p: Utilization, R=1 -p: probability processor idle

P: probability at least one task waiting and one server idle

P= Qi HN-1

Where Qi is probability that I servers is idle, and HN-1 is probability that the set of (N-i) servers are not idle and at least one has a task waiting

Qi = Ri, HN-1 = (1-R)N-I – [(1-R)R]N-i

P=1-(1-R)N(1-RN)-RN(2-R)N

R = 100 – 60 = 40% = 0.4

N = 20

P = 1-(0.6)20(1-(0.4)20)-(0.4)20(1.6)20 = 0.99983