



CSI3344 Principles of Distributed Systems

Workshop Solution 09

- Q1. What are the permissible delivery orderings for the combination of FIFO and total-ordered multicasting in the following figure (also see Figure. 8-15 on page 352 in the Textbook)?

Process P1	Process P2	Process P3	Process P4
sends m1	receives m1	receives m3	sends m3
sends m2	receives m3	receives m1	sends m4
	receives m2	receives m2	
	receives m4	receives m4	

A: There are six orderings possible:

Order 1	Order 2	Order 3	Order 4	Order 5	Order 6
m1	m1	m1	m3	m3	m3
m2	m3	m3	m1	m1	m4
m3	m2	m4	m2	m4	m1
m4	m4	m2	m4	m2	m2

- Q2. The printers of the same brand may produce random errors in some exceptional conditions. If such printers are used to build up a local printer server that should guarantee that at any time there are at least two printers are running in correct status, how many such printers are required to make up this printer server?

A: The type of failures is Arbitrary/Byzantine. Therefore the number of replicas needed formula should be $2k+1$ (where k is the number of printers that always need to be in correct status). So in order to satisfy the condition stated in the question, we will need $2*2+1 = 5$ printers.

Q3. The following three cases for multicasting can be classified into more than four multicasting schemes. Try to classify them and explain your classification.

(a)

Process 1	Process 2	Process 3
Sends m1	Receives m1	Receives m2
Sends m2	Receives m2	Receives m1

A: Unordered multicasts: message can be received in any order

(b)

Process 1	Process 2	Process 3	Process 4
Sends m1	Receives m1	Receives m3	Sends m3
Sends m2	Receives m3	Receives m1	Sends m4
	Receives m2	Receives m2	
	Receives m4	Receives m4	

A: FIFO-ordered multicasts: message received in the same order as they were sent from the same process, i.e., m1 before m2 and m3 before m4; and

Causal-ordered multicasts: message received in the same order as they were casually ordered (in the same process), i.e., m1 before m2 and m3 before m4.

(c) * P1 always has the priority.

Process 1*	Process 2	Process 3	Process 4
Sends m1	Receives m1	Receives m1	Sends m3
Sends m2	Receives m2	Receives m2	Sends m4
	Receives m3	Receives m3	
	Receives m4	Receives m4	

A: Total-ordered multicasts: message from P1 should be placed on top as their sending order, followed by messages sent from other processes (P4) in their sending order.

We may observe that, in addition to the priority given to the process1, this case is a combination of casual-ordered multicast and a FIFO-ordered multicast. Casual ordering delivers messages so that potential causality between different messages is preserved. In other words, if a message m1 causally precedes (or has higher priority than) another message m2, it is enforced to be delivered before m2 regardless of whether they were multicast by the same sender (or the same process). Therefore the case can also be classified into FIFO- and causal-ordered multicasting scheme.

END OF THE WORKSHOP SOLUTION