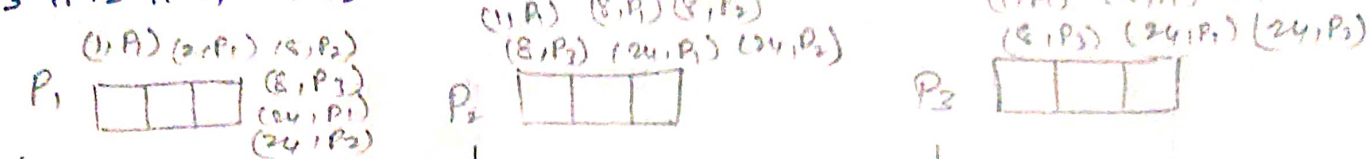
Distributed Systems

Nitheesh Kumar
205001070

Lamport's DM Algorithm

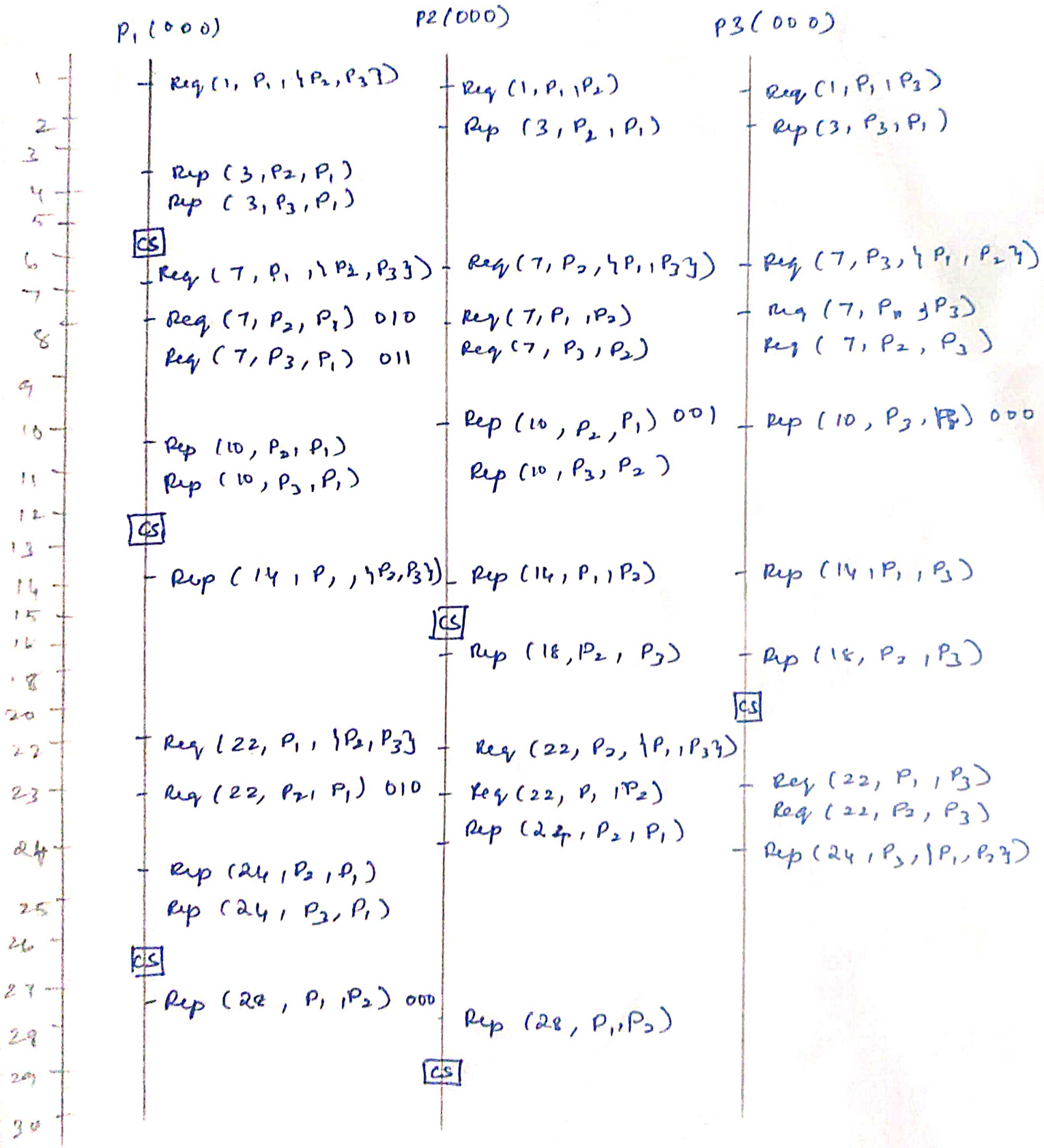
$P_1 \rightarrow P_3 \,||\, P_2 \,||\, P_1 \rightarrow P_2 \,||\, P_1$



	P_1	P_2	P_3
1	Req (1, P_1, {P_2, P_3}) add to queue	Req (1, P_1, P_2)	Req (1, P_1, P_3)
2		Rep (3, P_2, P_1)	Rep (3, P_3, P_1) add
3	Rep (3, P_2, P_1)		
4	Rep (3, P_3, P_1)		
5			
6	CS		
7	Rel (6, P_1, {P_1, P_2, P_3}) remove	Rel (6, P_1, P_2) remove	Rel (6, P_1, P_3) remove
8	Req (8, P_1, {P_2, P_3}) add	Req (8, P_2, {P_1, P_3}) add	Req (8, P_3, {P_1, P_2})
9			
10	Req (8, P_2, P_1) add Req (8, P_3, P_1) add	Req (8, P_1, P_2) add Req (8, P_3, P_2) add	Req (8, P_1, P_3) add Req (8, P_2, P_3) add
11		Rep (11, P_2, P_1)	Rep (11, P_3, P_1)
12	Rep (11, P_2, P_1)		
13	Rep (11, P_3, P_1)		
14			
15	CS Rel (15, P_1, {P_2, P_3}) remove	Rel (15, P_1, P_2) remove	Rel (15, P_1, P_3) remove
16			
17		CS	
18	Rel (18, P_2, P_1) remove	Rel (18, P_2, {P_1, P_3}) remove	Rel (18, P_2, P_1)
19			
20			CS Rel (21, P_3, {P_1, P_3}) remove
21	Rel (21, P_3, P_2) remove	Rel (21, P_3, P_2) remove	
22			
23	Req (24, P_1, {P_2, P_3}) add	Req (24, P_2, {P_1, P_3}) add	Req (24, P_1, P_3) add
24	Req (24, P_2, P_1) add	Req (24, P_1, P_2) add	Req (24, P_2, P_3) add
25		Rep (26, P_2, P_1)	Rep (26, P_3, P_1)
26	Rep (26, P_2, P_1)		
27	Rep (26, P_3, P_1)		
28			
29	CS Rel (29, P_1, {P_2, P_3}) remove	Rel (29, P_1, P_2) remove	Rel (29, P_1, P_3) remove
30			
31		CS	
32	Rel (32, P_2, P_1) remove	Rel (32, P_2, {P_1, P_3}) remove	Rel (32, P_2, P_3) remove

Lamport's algo. achieves mutual exclusion by sending Req, Rep and Rel messages. Hence 3(N-1) messages are required.

Ricart Agarwala Algorithm



This algorithm uses only req and rep messages to communicate thereby reducing to 2(N-1) messages ~~sending~~ by checking which replies have been deferred.