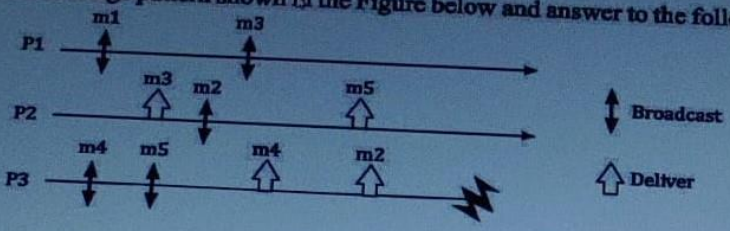
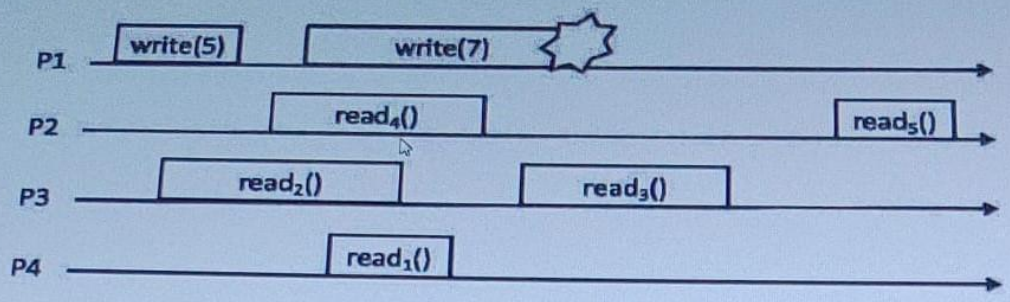
Ex 1: Provide the specification of the regular consensus primitive and describe the implementation presented during the lectures in synchronous systems. Finally, discuss its performance in terms of messages exchanging to reach consensus.

Ex 2: Consider the message pattern shown in the Figure below and answer to the following questions:



1. Complete the execution in order to have a run satisfying Reliable Broadcast but not Uniform Reliable Broadcast.
2. Provide all the delivery sequences satisfying causal order and total order.
3. Provide all the delivery sequences violating causal order and satisfying TO(UA, WNUTO) but not satisfying TO(UA, SUTO).

Consider the partial execution depicted in the Figure



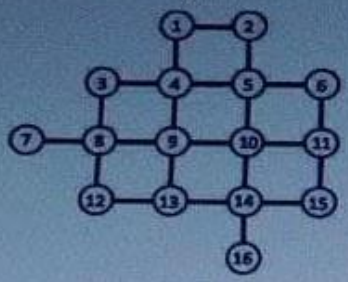
o the following questions:

efine ALL the values that can be returned by read operations (Rx) assuming that the
fers to a regular register.

fine ALL the values that can be returned by read operations (Rx) assuming that the
ers to an atomic register.

ign to each read operations (Rx) a return value that makes the execution linearizable.

Ex 5: Consider a distributed system constituted by n processes $\Pi=\{p_1, p_2 \ldots p_n\}$ with unique identifiers that exchange messages through FIFO perfect point-to-point links and are connected through a grid (i.e., each process p_i can exchange messages only with processes located at *nord*, *sud*, *east* and *west* when they exists).
An example of such network is provided in the following figure:



1. Write the pseudo-code of an algorithm implementing a Reliable Broadcast communication primitive assuming that processes may crash and each process is equipped with a perfect oracle, having the interface *new_neighbour(p, side)*, reporting a new neighbor p located in the direction *side* when the previous one is failing.
2. Let us assume now that k processes may fail experiencing Byzantine failure. Discuss which is the maximum value of k that keeps your solution correct.