5 Concurrent and Distributed Systems (tlh20)

A system is being designed to measure traffic flow into a city. Part of this system is a set of monitoring nodes, each using sensors to detect when a vehicle passes the monitoring node, and incrementing a per-node counter. The nodes communicate over a network to provide a city-wide total.

(a) Define each of the terms:

(i) Fair-loss network links. [1 mark]

(ii) Crash-recovery execution. [1 mark]

(iii) Asynchronous timing. [1 mark]

(b) Consider a version of the system using quorum-based replication. There is a fixed set of 5 nodes, meaning that each node holds a 5-tuple comprising the node’s most recent values for each of the replicated counters. A node should be able to operate if it can communicate with at least 2 other nodes. Describe how each of the following operations can be implemented by sending messages between nodes:

(i) A \texttt{setCount(n)} function to update the node’s local counter to \( n \) and to replicate the change to a quorum. [5 marks]

(ii) A \texttt{getTotalCount()} function to return the total of the counters from all of the nodes. [5 marks]

You should describe the messages sent and received by each node, along with how a node updates its local 5-tuple with new information when it receives messages, and how a node determines that the operation is complete.

(c) Does your system provide linearizable behaviour? Either explain why your system is linearizable, or provide an example showing a non-linearizable result. [3 marks]

(d) Consider a second version of the system that provides strong eventual consistency and allows the operations to always complete irrespective of the number of nodes available for communication. Summarize the changes needed to provide this behaviour. [4 marks]