Mandatory Exercise 2

Thomas Hoffmann Kilbak (thhk)
Kristoffer Bruun Højelse (krbh)
Jonas Glerup Røssum (jglr)
Emil Jäpelt (emja)
Group AD

September 28th 2020

A

Which of these times should it use to set its clock?

Round-trip (ms): 20 ms, Time (hr:min:sec): 10:54:28.342

To what time should it set it?

$t + T_{round}/2$

10:54:28.342 + 00:00:00.020 / 2

Estimate the accuracy of the setting with respect to the server's clock. If it is known that the time between sending and receiving a message in the system concerned is at least 8 ms, do your answers change?

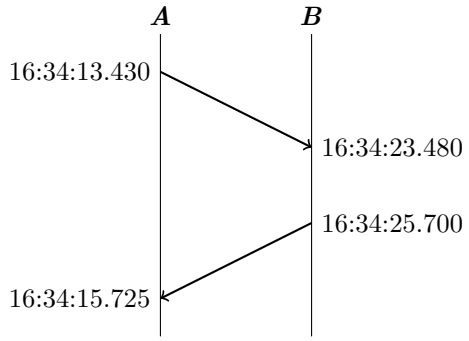
$t + T_{round}/2$

10:54:28.342 + 00:00:00.020 / 2

$\pm(T_{round}/2 - min)$

$\pm(20ms/2 - 8ms) = \pm 2ms$

B



Hours and minutes are truncated.

$$offset = \frac{t_{i-2} - t_{i-3} + t_{i-1} - t_i}{2} = \frac{25.7s - 15.725s + 23.48s - 13.43s}{2} = 10.0125s$$

$$accuracy = t_{i-2} - t_{i-3} + t_i - t_{i-1} = 23.48s - 13.43s + 15.725s - 25.7s = 0.075s$$

Formulas from: Distibuted Systems 5th ed. Coulouris - p606

C

C.1

Write out the happens-before relation in the following diagram.

$$p_1 \rightarrow p_2$$

$$q_1 \rightarrow q_2$$

$$r_1 \rightarrow r_2$$

$$p_1 \rightarrow q_2$$

$$q_1 \rightarrow p_2$$

$$q_1 \rightarrow r_4$$

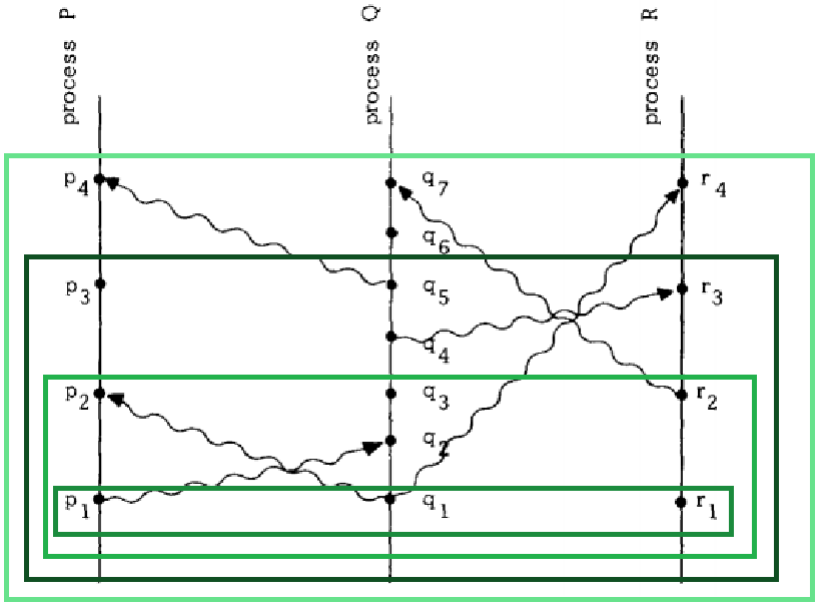
$$q_4 \rightarrow r_3$$

$$q_5 \rightarrow p_4$$

$$r_2 \rightarrow q_7$$

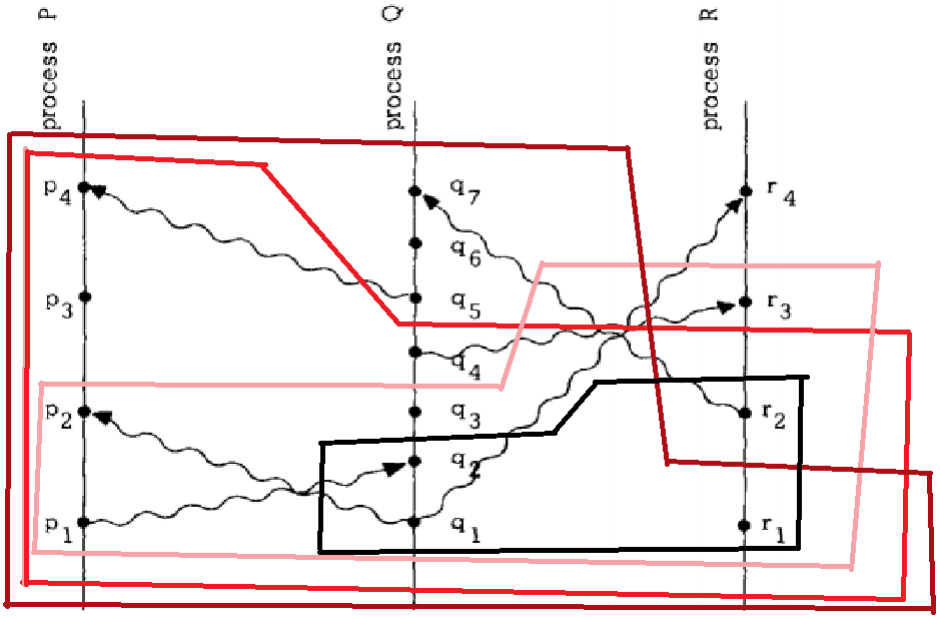
C.2

Identify 4 consistent cuts.



C.3

Identify 4 inconsistent cuts.



C.4

Write 2 different linearisations of the events in this diagram.

$$p_1 \rightarrow q_2 \rightarrow q_3 \rightarrow q_4 \rightarrow q_5 \rightarrow p_4$$

$$q_1 \rightarrow r_4$$

D

D.1

For example, if one process is in uneven state, and two processes are in even states, and one of the even states transmits a message. Meanwhile the other even proces transmits a message while the first message is in transit. One message arrives at each even proces, and they both generate uneven numbers.

D.2

There are four possible inital states: 0, 1, 2 and 3 active processes. All of these may lead to a deadlock. If 0 are active, it starts in a deadlock. States with 1 or 3 active processes can transpose into a 2-active-processes-state like above.

In full: If one is active, it may send out a message that causes an even number, which leads to there being two active processes. Then it can deadlock as described in the previous exercise. For two processes, it can deadlock as previously described. And for 3 processes, one process may send a message that causes an odd number, putting the system in a state like in the previous exercise.

D.3

The algorithm makes every process record its own state, and if every local state is odd, the system is deadlocked.

D.4

The process with state 3 start by sending a marker message to every process. When the two other processes with state 4 and 7 receives a marker message, the processes records their local state, as well as starts to record every incoming message from then on, until the time where the sender recorded its state.

It is not possible to conclude that the system has deadlocked, as it depends on whether odd or even numbers have been generated.

D.5

The process with state 3 start by sending a marker message to every process. When the two other processes with state 5 and 7 receives a marker message, the processes records their local state, as well as starts to record every incoming message from then on, until the time where the sender recorded its state.

It is possible to conclude that the system is deadlocked, as it is impossible for any of the processes to be active, as they are all odd.

E

Mallory would still be able to tamper with the messages. If she intercepts a message, she could change the contents and change the checksum to match the new message, so the checksum would not make the message more trustworthy. Even if Mallory was not able to change the checksum, the check would still be somewhat useful. In this case, Mallory would still be able to tamper with the messages, but Bob and Alice would know that the message was either corrupted or had been tampered with.

To accomplish tamperproofing, they would need to be able to verify the sender of the message using asymmetric encryption to produce a signature or encrypt the message.