ECE 428 / CS 425 / CSE 424 Homework 1 Solutions

(a) 11-4 The third time request has the shortest round-trip-time, thus it is the best candidate for clock synchronization. The accuracy is RTT/2 = 10ms. If it is known that the time between sending and receiving a message is at least 8ms, then the time in doubt reduces to 4ms, hence a 2ms accuracy.

(b) 11-7.

$$T_i = 16:34:13.430,\quad T_{i-1} = 16:34:23.480,\quad T_{i-2} = 16:34:25.7,\quad T_{i-3} = 16:34:15.725$$

$$accuracy := \frac{d_i}{2} = \frac{T_{i-2} - T_{i-3} + T_i - T_{i-1}}{2} = 0.0375\,s$$

$$offset := o_i = \frac{T_{i-2} - T_{i-3} + T_{i-1} - T_i}{2} = 10.0125\,s$$

(c) 11-10.

If e and e_0 are connected by a message m, then $L(e_0) = L(e) + 1$ or $L(e_0) = \lambda + 1$, where λ is the internal Lamport clock of the process receiving message m, and $\lambda > L(e)$ by LC2a and LC1, thus $L(e_0) > L(e)$.
If e_{i-1} and e_i are connected by message m_j, then $L(e_i) > L(e_{i-1})$ by the same above reasoning.
If e and e' are connected by n messages in the run $\{e, e_0, \ldots, e_{n-2}, e'\}$, then by the above reasoning, $L(e') > L(e_{n-2}) > \ldots > L(e_0) > L(e)$ and thus,

$L(e') > L(e)$. ☺

(d)

$L(e_1) = L(e_2) = L(e_8) = 1$
$L(e_3) = 2$
$L(e_5) = L(e_4) = 3$
$L(e_6) = L(e_{10}) = L(e_{13}) = 4$
$L(e_7) = 5$
$L(e_9) = 6$
$L(e_{11}) = 7$
$L(e_{12}) = L(e_{14}) = 8$

(e)

Event e_i	***Vector clock $V(e_i)$***
1	(1,0,0,0)
2	(0,0,1,0)
3	(2,0,0,0)
4	(2,1,0,0)
5	(3,0,0,0)
6	(2,2,0,0)
7	(2,2,2,0)
8	(0,0,0,1)
9	(2,2,3,1)
10	(3,0,0,2)
11	(2,2,4,1)

Event e_i	***Vector clock*** $V(e_i)$
12	(2,3,4,1)
13	(4,0,0,0)
14	(4,2,5,1)

(f) The cut C1 is consistent because the only messages crossing the line of the cut are sent from events on the left of the cut. The cut C2 is inconsistent because the message passed from e_{11} to e_{12} breaks the causal relationship by including the receiving event e_{12} and excluding the sending event e_{11}.

(g) Events that happened before e_8 according to Lamport clocks: $\{\varnothing\}$
Events concurrent to e_8: $\{e_1, e_2, e_3, e_4, e_5, e_6, e_7\}$

(h) 11.14
There are two cases of the operation of the snapshot algorithm in this example.

Case 1: P records its state as 101, sends a marker message to Q and begins recording the state of its channel with Q. Process Q sends m before receiving the snapshot marker message from P. Then Q receives the marker from P, records its state as 102, records the channel state as the empty set, and sends a marker message back to P. P records the state of the channel as containing the message m.

Case 2: P records its state as 101, sends a marker message to Q and begins recording the state of its channel with Q. Process Q receives the message m from P and updates its state to 102. Q then receives the marker message from P, records its state as 102, records the channel state as the empty set, and sends a marker message to P. P records the state of the channel as the empty set. Q then sends the message m.