

Esercizi su sincronizzazione dei clock

Corso di Sistemi Distribuiti e Cloud Computing A.A. 2023/24

Valeria Cardellini

Laurea Magistrale in Ingegneria Informatica

Esercizio 1

Si supponga che un client usi l'algoritmo di Cristian per sincronizzarsi con un time server e che i valori del round-trip time e del timestamp restituito dal time server siano i seguenti:

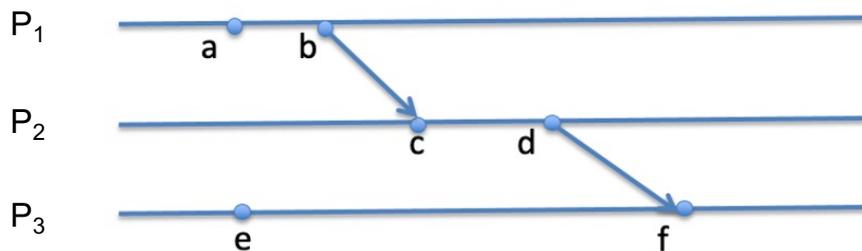
Round-trip time (ms)	Time (hr:min:sec:msec)
22	10:54:23.674
25	10:54:25.450
20	10:54:28.342

Come imposta il client il suo clock in modo da massimizzare l'accuratezza? Motivare la risposta.

Il client imposta il suo clock fisico a $10:54:28.342 + \text{RTT}/2$ quindi pari a $10:54:28.352$, dal momento che seleziona il valore del clock del time server corrispondente al minimo RTT.

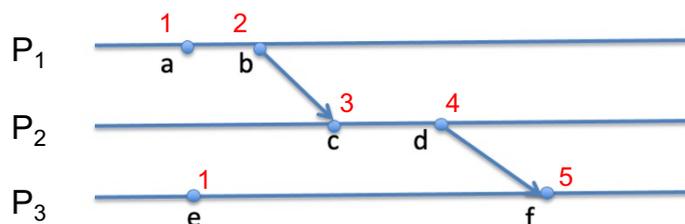
Esercizio 2

- a) Con riferimento al diagramma temporale sottostante, calcolare il clock logico scalare e vettoriale di tutti gli eventi da a ad f.
- b) Si discuta se, dato il valore del clock sia scalare sia vettoriale determinato al punto a), si può affermare che $c \parallel e$



Esercizio 2: soluzione

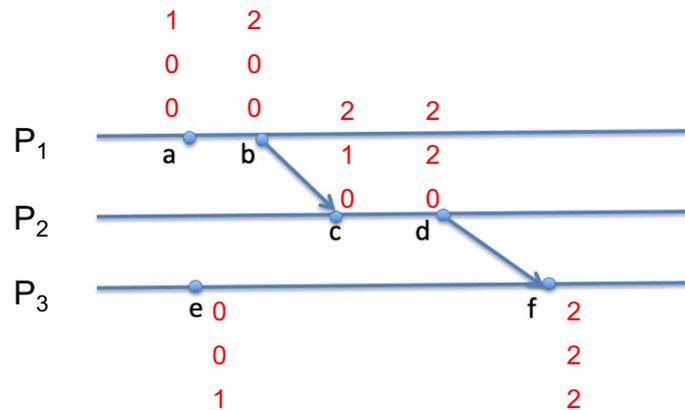
- Clock logico scalare:



Confrontando il valore del clock logico scalare degli eventi c ed e , si può solo concludere che $c \not\rightarrow e$ essendo $3 > 1$ ma non si può concludere se $e \rightarrow c$ oppure $c \parallel e$

Esercizio 2: soluzione

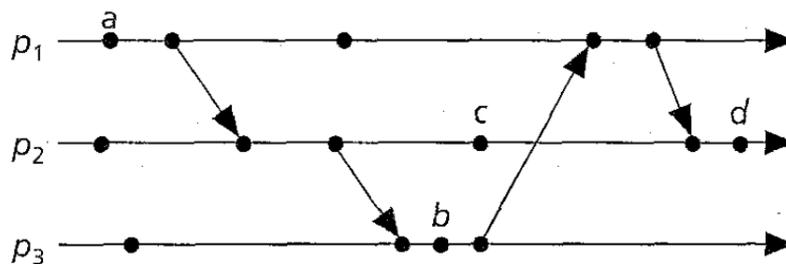
- Clock logico vettoriale:



Confrontando il valore del clock logico vettoriale degli eventi c ed e, si può concludere che $c \parallel e$ essendo $\text{not}((2 \ 1 \ 0) < (0 \ 0 \ 1))$ and $\text{not}((0 \ 0 \ 1) < (2 \ 1 \ 0))$

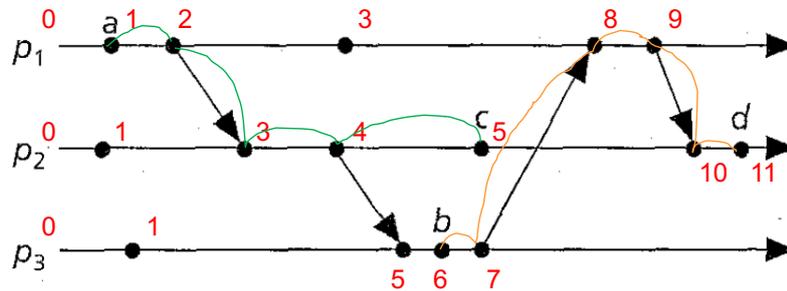
Esercizio 3

- Con riferimento al diagramma temporale sottostante, calcolare il clock logico scalare e vettoriale di tutti gli eventi.
- In base ai valori del clock scalare e di quello vettoriale, si discuta se le coppie di eventi (a, c) (b, c) e (b, d) sono in relazione happened-before oppure no motivando opportunamente la risposta



Esercizio 3: soluzione

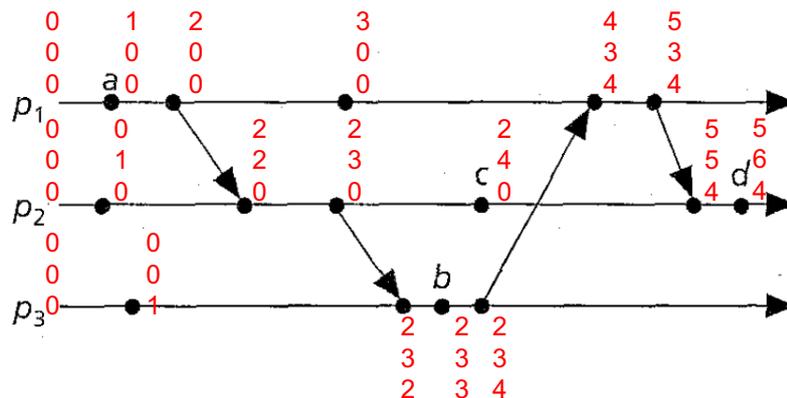
- Clock logico scalare:



- Confrontando il valore del clock logico scalare degli eventi, si può solo concludere che:
 - $c \Rightarrow a$, $b \Rightarrow c$, $d \Rightarrow b$
- Applicando le proprietà della relazione happened-before:
 - $a \rightarrow c$ (sequenza verde), $b \parallel c$, $b \rightarrow d$ (sequenza arancione)

Esercizio 3: soluzione

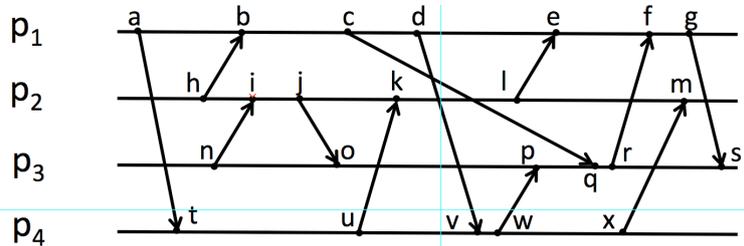
- Clock logico vettoriale:



- Confrontando il valore del clock logico vettoriale degli eventi, si può concludere che:
 - $a \rightarrow c$ essendo $(1\ 0\ 0) < (2\ 4\ 0)$
 - $b \parallel c$ essendo $\text{not}((2\ 3\ 3) < (2\ 4\ 0))$ and $\text{not}((2\ 4\ 0) < (2\ 3\ 3))$
 - $b \rightarrow d$ essendo $(2\ 3\ 3) < (5\ 6\ 4)$

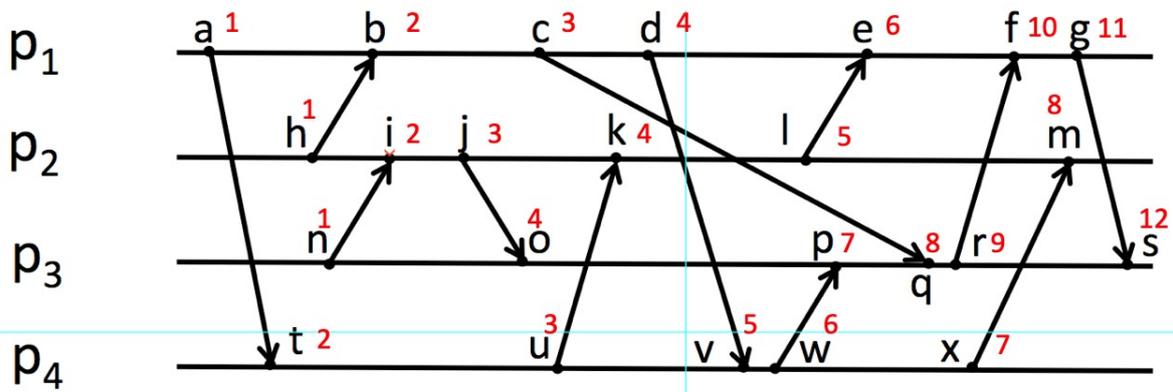
Esercizio 4

- Definire la relazione happened-before introdotta da Lamport e spiegare come si rappresentano i clock logici scalare e vettoriale.
- Descrivere i relativi protocolli di aggiornamento dei clock e indicare un algoritmo distribuito esaminato a lezione in cui viene usato il clock logico scalare ed un algoritmo in cui viene usato quello vettoriale.
- Con riferimento al diagramma temporale sottostante: Calcolare il clock logico scalare di tutti gli eventi da *a* a *x*. Calcolare il clock logico vettoriale di tutti gli eventi da *a* a *x*. In base ai valori calcolati, può essere identificata una violazione della causalità? Spiegare perché.



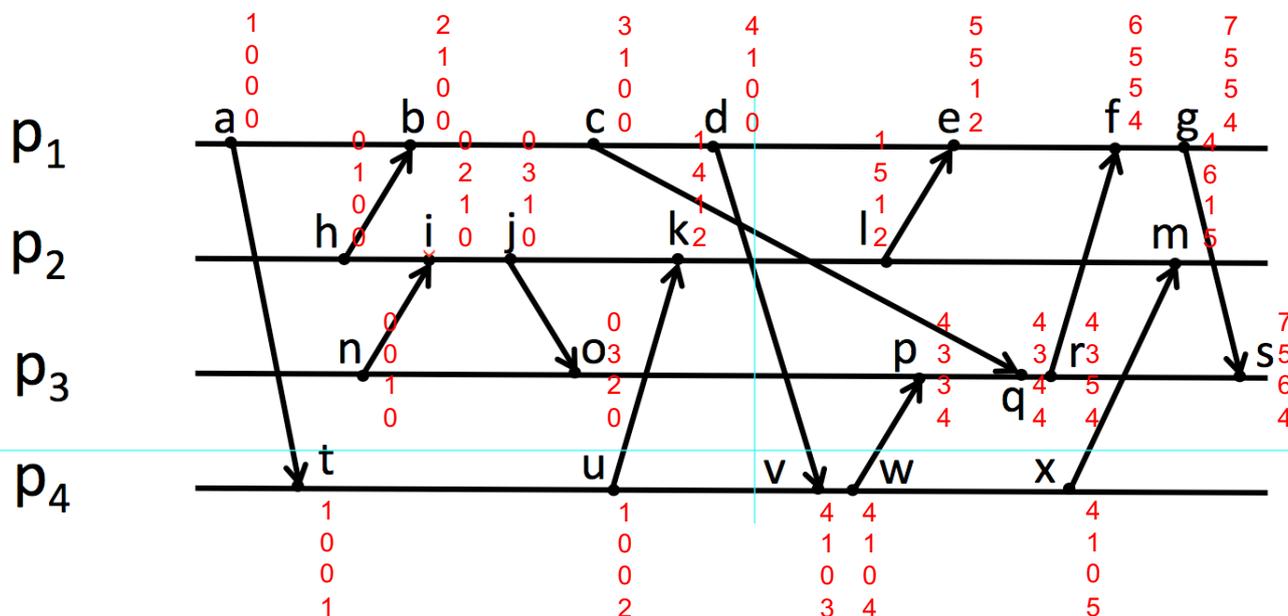
Esercizio 4: soluzione c)

- Clock logico scalare:



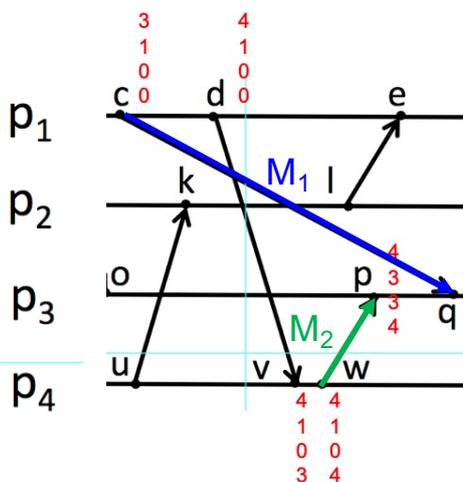
Esercizio 4: soluzione c)

- Clock logico vettoriale:



Esercizio 4: soluzione c)

- Violazione di causalità: gli eventi coinvolti sono c, p e q (p_3 riceve prima p e poi q)



- p_3 può identificare la violazione di causalità usando il clock vettoriale
- Il messaggio M_1 inviato da p_1 ha timestamp (3 1 0 0)
- Il messaggio M_2 inviato da p_4 ha timestamp (4 1 0 4)
- (3 1 0 0) < (4 1 0 4): un processo che ha già visto c (ovvero p_4) ha inviato un messaggio a p_3 prima che p_3 abbia ricevuto M_1
- Quindi p_3 si accorge di aver ricevuto prima l'effetto e poi la causa dal confronto dei due timestamp: quello di M_1 (ricevuto da p_3 dopo M_2) è minore del timestamp di M_1