Sistemas Distribuídas - 1ªChamada

IST - LEIC-A/ LEIC-T/ LETI - 2021-2022
2 de maio de 2022

- A classificação máxima é de 20 pontos.
- A classificação mínima para aprovação é de 9 valores.
- Todas as respostas devem se dadas na "Folhas de Respostas".
- Identifique com o seu número e nome todas as folhas de resposta.
- Não pode sair da sala durante a primeira hora do exame.
- A utlização de telemóveis ou de equipamentos informáticos durante o exame é proibida.
- Nas respostas erradas às perguntas de escolha múltipla é descontada a cotação da pergunta dividida pelo número de alternativas.
- O exame tem a duração de 2 horas.

Chamada a Procedimentos Remotos

Considere um serviço com a seguinte especificação:

```
syntax = "proto3";
package sd;

message IncrementRequest {
}

message IncrementResponse {
}

message ReadRequest {
}

message ReadResponse {
    int32  countervalue = 1;
}

service Counter {
    rpc increment(IncrementRequest) returns (IncrementResponse); // increments by one
    rpc read(ReadRequest) returns (ReadResponse);
}
```

Questão 1 (*1 valor*) Apenas a partir desta especificação, diga se seria possível:

- Prever o formato da mensagem enviada na rede para invocar pedido de "`IncrementRequest`".
- Prever o formato em memória da variavel "`countervalue`" no servidor.
- Prever o formato em memória da variavel "`countervalue`" no cliente.

Questão 2 (*1 valor*) Considere que o servidor possui o contador com o valor de 100. Considere que existe apenas 1 único cliente, que faz apenas uma única invocação do método `increment`. Diga qual ou quais os valores possíveis para o contador no servidor, após a invoção do método `increment`, para o caso em que o serviço de chamadas a procedimentos remotos oferece as seguintes semânticas:

- Pelo menos uma vez
- No máximo uma vez
- Exactamente uma vez

Sincronização de Relógios

Considere um algoritmo de sincronização de relógios baseado num servidor centralizado p_s (por exemplo, o algoritmo de Cristian). Neste serviço, os restantes processos fazem uma leitura remota ao servidor e ajustam o valor do seu relógio para o aproximar do valor do relógio no servidor. Considere ainda que tempos mínimos de envio de uma mensagem na rede não são conhecidos. Considere a seguinte execução:

leitura	origem p_i	*tempo de envio* (no relógio de p_i)	*tempo de recepção* (relógio de p_i)	*valor na resposta (relógio de p_s)*
leitura 1	p_1	100	106	101
leitura 2	p_2	102	112	110

Assuma que o erro introduzido pelo desvios dos relógios durante a sincronização pode ser descartado.

Questão 3 (*1 valor*) Para cada uma das leituras, diga qual é o ajuste que cada cliente faz ao seu próprio relógio e qual o erro dessa leitura.

Questão 4 (*1 valor*) Qual é a diferença máxima entre o relógio dos dois clientes no final da sincronização.

Relógios Lógicos

Considere a execução ilustrada na Figura 1.

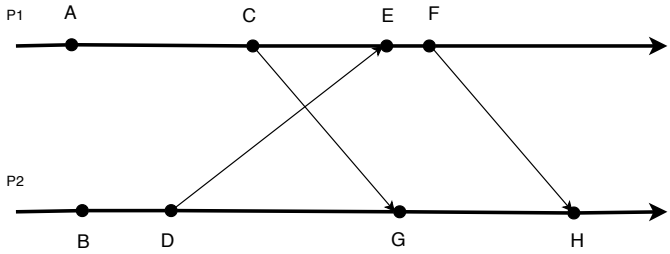


Figura 1: Execução distribuída

Questão 5 (*1 valor*) Considere que usa relógios lógicos de Lamport para marcar todos os eventos (isto é, tanto os eventos de emissão como os eventos de recepção de mensagens). Assuma que os eventos A e B foram marcados com o seguinte valores de tempo lógico: $l(A) = 10$ e $l(B) = 12$. Qual é o valor do relógio atribuído ao evento F?

Questão 6 (*1 valor*) Considere que usa relógios vectoriais para marcar todos os eventos (isto é, tanto os eventos de emissão como os eventos de recepção de mensagens). Assuma que os eventos A e B foram marcados com o seguintes relógios vectoriais: $vector(A) = [2, 1]$ e $vector(B) = [1, 4]$. Qual é o valor do relógio vectorial atribuído ao evento H?

Gossip - Lazy Replication

Considere o sistema replicado conhecido por "Lazy Replication"ou "Gossip", no qual as operações são propagadas "nos bastidores" por propagação epidémica. Considere um sistema com 3 réplicas, em que o estado de cada réplica é capturado por um relógio vectorial. Considere que num dado instante, os servidores encontram-se no seguinte estado: S_1 =(2, 6, 2), S_2 = (1, 6, 7) and S_3 = (2, 6, 7).
Considere também um cliente, cujo estado é representado pelo seguinte vector: *prev*=(2, 5, 5).

Questão 7 (*0.5 valor*) Que servidores poderiam servir um pedido de leitura deste cliente imediatamente? Qual o valor do relógio do cliente após a leitura?

Questão 8 (*0.5 valor*) Que servidores poderiam servir um pedido de escrita deste cliente?

Exclusão Mútua

Questão 9 (*1 valor*) Sobre os algoritmos de exclusão mútua e suas propriedades, qual das seguintes afirmações é falsa?

1. Um algoritmo assegura *safety* quando nunca é possível estar mais do que um processo na secção crítica.
2. O algoritmo centralizado não é tolerante a falhas do coordenador.
3. No algoritmo distribuído de Ricart e Agrawala a entrada de um processo na sua zona critica requer n rondas de troca de mensagens num sistema com n processos.
4. No algoritmo distribuído de Ricart e Agrawala usam-se relógios lógicos para garantir *safety*.
5. O algoritmo distribuído de Ricart e Agrawala evita o interbloqueio (*deadlock*).

Eleição de Líder

Considere um sistema de 5 processos, $\{p_1, p_2, p_3, p_4, p_5\}$ que escolhem um líder usando o algoritmo de "Bully". Considere que p_5 é o líder e falha.

Questão 10 (*1 valor*) Considere que o processo p_2 é o primeiro a suspeitar da falha do líder. Neste caso:

- Para que processos envia p_2 uma mensagem de ELECTION?
- Considere que p_3 recebe a mensagem ELECTION vinda de p2. Que mensagens são enviadas de seguida por p_3?
- No final da execução do algortimo, qual o processso que envia uma mensagem de COORDINATOR?

Salvaguardas

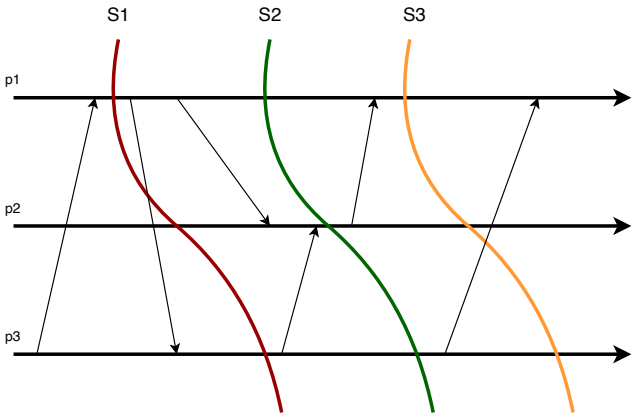


Figura 2: Execução distribuída

Questão 11 (*1 valor*) Considere a execução ilustrada na Figura 2. Para cada um dos cortes $S1$, $S2$ e $S3$, diga se capturam um estado incorente, fracamente coerente, ou fortemente coerente.

Questão 12 (*1 valor*) Considere a execução ilustrada na Figura 3, onde cada processo possui n tokens (inicialmente 100) e cada mensagem transfere 10 tokens entre dois processos. Considere que o processo p_3 inicia uma salvaguarda no instante X, executando o algoritmo de Chandy-Lamport. Qual vai ser o estado capturado pelo algoritmo?

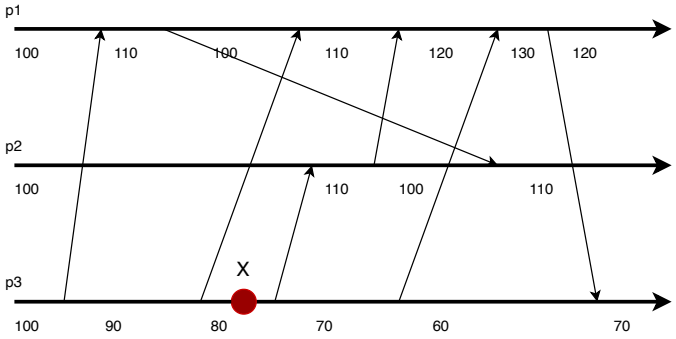


Figura 3: Execução distribuída

Registos

Considere um sistema de replicação que funciona da seguinte maneira. Um conjunto de réplicas é organizado numa cadeia, em que as escritas são aplicadas na cabeça (que atribuí um número de sequência à escrita) e propagadas na cadeia até à cauda, sendo uma resposta enviada ao cliente pela réplica da cauda, depois de todas as réplica terem sido actualizadas. Para distribuir a carga das leituras, os clientes sempre que fazem uma leitura, escolhem uma réplica aleatoriamente e leem dessa réplica.

Questão 13 (*1 valor*) Este sistema oferece registos atómicos? Justifique.

Questão 14 (*1 valor*) Que informação seria necessário manter no cliente para garantir que não havia leituras incoerentes pelo mesmo cliente, isto é, que o cliente lia sempre as suas escritas ("Read your writes")?

Espaços de Tuplos

O algoritmo de Xu-Liskov para concretizar um espaço de tuplos usa UDP e, em todas as operações, os pedidos são retransmitidos até que o número necessário de respostas seja recolhido. Considere a primeira fase do TAKE neste algoritmo, descrita na Figure 4.

take *Phase 1: Selecting the tuple to be removed*

1. The requesting site multicasts the *take* request to all members of the view;
2. On receiving this request, each replica acquires a lock on the associated tuple set and, if the lock cannot be acquired, the *take* request is rejected;
3. All accepting members reply with the set of all matching tuples;
4. Step 1 is repeated until all sites have accepted the request and responded with their set of tuples and the intersection is non-null;
5. A particular tuple is selected as the result of the operation (selected randomly from the intersection of all the replies);
6. If only a minority accept the request, this minority are asked to release their locks and phase 1 repeats.

Figura 4: Take, primeira fase

Questão 15 (*1 valor*) Suponha que substituía o uso de UDP por TCP. Ainda seria necessário repetir o passo 1? Por outras palavras, ainda seria necessário o passo 4?

Ordem Total

Considere o algoritmo para estabelecer uma ordem total inventado pelo Dale Skeen designado por "acordo colectivo". Neste algoritmo, cada receptor mantem uma fila ordenada de mensagens, em que cada entrada na fila é um tuplo com o seguinte formato:

⟨ id_da_mensagem, emissor, número_de_sequência, estado (**T**entativo ou **F**inal) ⟩.

Considere um sistema com três réplicas e vários clientes que enviam mensagens para estas réplicas usando o algoritmo de acordo colectivo. Considere que num dado instante, o estado das réplicas é o seguinte:

réplica		
r_1	r_2	r_3
entregues		
	$\langle A, c_1, 1, F\rangle$	$\langle A, c_1, 1, F\rangle$
pendentes		
$\langle A, c_1, 1, T\rangle$	$\langle E, c_5, 2, T\rangle$	$\langle D, c_4, 2, T\rangle$
$\langle B, c_2, 2, T\rangle$	$\langle D, c_4, 3, T\rangle$	
$\langle C, c_3, 3, T\rangle$		
$\langle D, c_4, 4, T\rangle$		

Questão 16 (*1 valor*) É possível prever quando é que a mensagem E vai ser entregue relativamente à mensagem D? Como resposta a esta questão, indique qual das seguintes afirmações está correcta:

1. D vai ser entregue antes de E, porque o número final de E será necessariamente superior ao de D.
2. E vai ser entregue antes de D, porque o número final de D será necessariamente superior ao de E.
3. É impossível de prever, pois depende de como for calculado o número final para cada mensagem.
4. Será E, porque E foi ordenada antes de D na réplica r_2
5. Vai depender do número final que for atribuído a B.

Consenso

Considere um sistema síncrono em que existe um tempo máximo Δ para a entrega de qualquer mensagem enviada por um processo que não falha. Considere o seguinte algoritmo para resolver o problema do consenso:

1. Cada processo envia o seu valor para todos os outros processos;
2. Todos os processos esperam o tempo máximo Δ
3. Cada processo escolhe o valor mínimo dos valores recebidos e faz output desse valor.

Questão 17 (*1 valor*) Qual das seguinte frases é verdadeira neste contexto?

1. Este algoritmo resolve o consenso, mas apenas em sistemas síncronos.
2. O algoritmo devia retornar o valor mais votado e não o mínimo.
3. O algoritmo deveria retornar o máximo e não o mínimo.
4. Para resolver o consenso, o algorimo precisaria de mais uma ronda, onde todos os processos enviariam aos outros processos o valor que tinham escolhido na primeira ronda.
5. Neste algoritmo, se o processo que propõe o valor mínimo falha, processos diferentes podem retornar valores diferentes, violando o consenso.

Transacções Distribuídas

Considere um participante no protocolo de confirmação atómica em duas fases (two-phase commit). O participante recebe o PREPARE do coordenador e confirma que pode fazer COMMIT à transacção (isto é, envia um OK) ao cooordenador. Depois disto, o coordenador falha.

Questão 18 (*1 valor*) Neste caso, o participante pode abortar a transacção? Justifique.

Segurança e Canais Seguros

Questão 19 (*1 valor*) Considere que um dado participante A quer enviar uma mensagem assinada a outro participante, e gera uma assinatura digital usando a sua chave privada e envia a mensagem e a assinatura para o destino da seguinte forma:

Alternativa 1: $\langle m, \{Digest(m)\}_{K_A^-} \rangle$

Considere uma alternativa em que A decide também cifrar m com a sua chave privada, enviando:

Alternativa 2: $\langle \{m\}_{K_A^-}, \{Digest(m)\}_{K_A^-} \rangle$

A segunda alternativa é mais segura, menos segura, ou igualmente segura que a primeira? A segunda alternativa é computacionalmente mais eficiente, menos eficiente, ou igualmente eficiente que a primeira?

Questão 20 (*1 valor*) Considere dois participantes que partilham um segredo K_s. A quer enviar uma mensagem m a B e garantir que B consegue detectar qualquer alteração à mensagem que ocorra durante a transmissão. Para isso, A calcula um *código de autenticação* da seguinte forma:

$\text{MAC}_m = Digest(m|K_s)$

e envia para B uma mensagem contendo:

$\langle m, \text{MAC}_m \rangle$

Considere que um adversário substitui m por m' e entrega a B a seguinte mensagem.

$\langle m', \text{MAC}_m \rangle$

Qual das seguintes operações deve B fazer para detectar este ataque?

1. Verificar se $\text{MAC}_m = Digest(m')$
2. Verificar se $\text{MAC}_m = Digest(m'|K_s)$
3. Verificar se $\text{MAC}_m = \{m'\}_{K_s}$
4. Verificar se $\text{MAC}_m = \{\text{MAC}_m\}_{K_s}$
5. Verificar se $\text{MAC}_m = \{m'|K_s\}_{K_s}$

Questão 21 (*1 valor*) Considere que A pretende estabeler um canal seguro com B. Neste contexto, A vai enviar um *nonce* a B. Qual das seguintes alternativas daria um *nonce* mais apropriado:

1. Uma leitura do valor do relógio de A.
2. O endereço de IP de A.
3. A chave pública de A.
4. A chave pública de B.
5. $\{K_B^+\}_{k_A^-}$

Folha de Respostas (1/4)

Número:
Nome:
Versão

§

Chamada a Procedimentos Remotos:

		Sim	Não
Questão 1	Prever o formato na rede		
	Prever o formato no servidor		
	Prever o formato no cliente		

		Opção 1	Opção 2	Opção 3	etc
Questão 2	Pelo menos uma vez				
	No máximo uma vez				
	Exactamente uma vez				

§

Sincronização de relógios:

	leitura 1	leitura 2
Questão 3	delta_1:	delta_2:
	erro_1:	erro_2:
Questão 4	diferença máxima :	

§

Relógios lógicos:

Questão 5	evento F:
Questão 6	evento H:

§

Lazy Replication

	Servidor	Pode servir o pedido (sim/não)?	valor do relógio (apenas se sim)
Questão 7	S_1		
	S_2		
	S_3		

	Servidor	Pode servir o pedido (sim/não)?
Questão 8	S_1	
	S_2	
	S_3	

§

Exclusão mútua:

Questão 9	

§

Eleição de líder:

Questão 10	p_2 envia ELECTION para:
	p_3 envia ELECTION para
	p_3 envia ANSWER para:
	p_3 envia COORDINATOR para:
	Quem envia COORDINATOR é:

§

Cortes coerentes:

Questão 11	S_1 é:
	S_2 é:
	S_3 é:

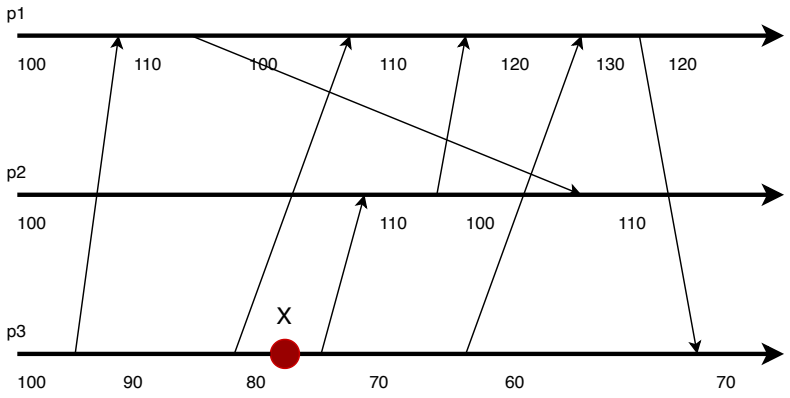
Folha de Respostas (2/4)

Número:
Nome:
Versão

§

Chandy-Lamport:

Questão 12	p_1:	p_2:	p_3:
	c_{11}: $\emptyset$	c_{12}:	c_{13}:
	c_{21}:	c_{22}: $\emptyset$	c_{23}:
	c_{31}:	c_{32}:	c_{33}: $\emptyset$
	(ilustre a execução na figura abaixo		



§

Registos:

Questão 13	
Questão 14	

Folha de Respostas (3/4)

Número:
Nome:
Versão

§

Espaço de Tuplos:

Questão 15	

§

Ordem Total:

Questão 16	

§

Consenso:

Questão 17	

§

Transacções:

Questão 18	

§

Segurança:

Questão 19	(marque as certas)			
	mais seguro:	menos seguro:	igualmente seguro:	
	mais eficiente:	menos eficiente:	igualmente eficiente:	
Questão 20				
Questão 21				

Folha de Respostas (4/4)

Número:
Nome:
Versão

Use esta página como rascunho

Folha de Respostas (1/4)

Número:
Nome:
Versão

§

Chamada a Procedimentos Remotos:

		Sim	Não
Questão 1	Prever o formato na rede	×	
	Prever o formato no servidor		×
	Prever o formato no cliente		×

		Opção 1	Opção 2	Opção 3	etc
Questão 2	Pelo menos uma vez	101	102	103	etc
	No máximo uma vez	100	101		
	Exactamente uma vez	101			

§

Sincronização de relógios:

Questão 3	leitura 1		leitura 2	
	delta_1:	-2	delta_2:	$+3$
	erro_1:	± 3	erro_2:	± 5
Questão 4	diferença máxima :			8

§

Relógios lógicos:

Questão 5	evento F:	15
Questão 6	evento H:	[5,7]

§

Lazy Replication

	Servidor	Pode servir o pedido (sim/não)?	valor do relógio (apenas se sim)
Questão 7	S_1	não	
	S_2	não	
	S_3	sim	[2,6,7]

	Servidor	Pode servir o pedido (sim/não)?
Questão 8	S_1	sim
	S_2	sim
	S_3	sim

§

Exclusão mútua:

Questão 9	3

§

Eleição de líder:

Questão 10	p_2 envia ELECTION para:	$\{p_3, p_4, p_5\}$
	p_3 envia ELECTION para	$\{p_4, p_5\}$
	p_3 envia ANSWER para:	$\{p_2\}$
	p_3 envia COORDINATOR para:	$\emptyset$
	Quem envia COORDINATOR é:	p_4

§

Cortes coerentes:

Questão 11	S_1 é:	incoerente
	S_2 é:	coerente (fortemente)
	S_3 é:	coerente (fracamente)

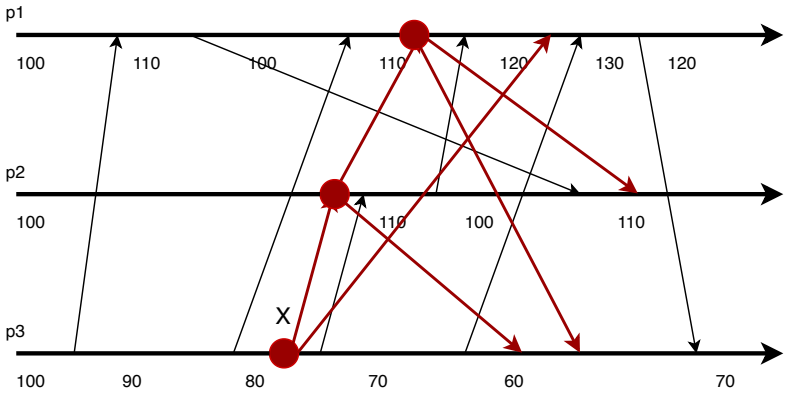
Folha de Respostas (2/4)

Número:
Nome:
Versão

§

Chandy-Lamport:

Questão 12	p_1:	110	p_2:	100	p_3:	80
	c_{11}:	$\emptyset$	c_{12}:	10	c_{13}:	$\emptyset$
	c_{21}:	$\emptyset$	c_{22}:	$\emptyset$	c_{23}:	$\emptyset$
	c_{31}:	$\emptyset$	c_{32}:	$\emptyset$	c_{33}:	$\emptyset$
	(ilustre a execução na figura abaixo					



§

Registos:

Questão 13	Não Devido ao tempo que demora a propagar a versão na cadeia, um cliente por ler uma versão nova (por exemplo da cabeça) e posteriormente outro cliente pode ler a versão anterior (por exemplo da cauda).
Questão 14	Nada. Este algoritmo já assegura "read your writes" (note-se que não assegura "monotonic reads")

Folha de Respostas (3/4)

Número:
Nome:
Versão

§

Espaço de Tuplos:

Questão 15	Sim, o passo 4 continuava a ser necessário. Isto porque não basta que todas as réplicas recebem o pedido de TAKE. É preciso encontrar um tuplo que esteja em todas as réplicas.

§

Ordem Total:

Questão 16	1 O número final da mensagem D será $4 = MAX(4, 3, 2)$ O número final de E será o máximo dos números "tentativos", mas necessariamente superior a 4 pois quando E chegar a r_1 ficará com um número 5 ou superior

§

Consenso:

Questão 17	5

§

Transacções:

Questão 18	Não. O participante não sabe se o coordenador enviou o COMMIT antes de falhar. Tem de esperar até o coordenador recuperar.

§

Segurança:

Questão 19	(marque as certas)		
	mais seguro:	menos seguro:	igualmente seguro: ×
	mais eficiente:	menos eficiente: ×	igualmente eficiente:
Questão 20	2		
Questão 21	1		

Folha de Respostas (4/4)

Número:
Nome:
Versão

Use esta página como rascunho