

Sistemas Distribuídos, 2016/17 - 2º MINI Teste

Todas as perguntas têm a mesma cotação. Escolha apenas uma resposta em cada alínea.

Cada resposta de escolha múltipla errada desconta 1/4 da sua cotação.

No caso de encontrar mais do que uma resposta certa, escolha a que faz a afirmação mais forte.

Número: _____ Nome: _____

1. No RPC existem várias semânticas para tolerar faltas da comunicação. Considere o seu uso na execução de um RPC entre um cliente e um servidor.
- A. Na semântica exatamente-uma-vez tolera qualquer tipo de falta da rede.
 - B. Na semântica exatamente-uma-vez tolera faltas silenciosas do nó.
 - C. Na semântica exatamente-uma-vez só permite tolerar faltas temporárias da rede.
 - D. Na semântica exatamente-uma-vez tolera faltas arbitrárias do nó.

2. O cliente de um sistema de primary-backup (funcionando de acordo com os pressupostos deste protocolo) executa uma operação de leitura e escrita sobre um objecto X com os seguintes resultados:
read (x) -> 3; write (x) -> 5; read (x) -> 3.

Assuma que, enquanto as operações acima são invocadas, nenhum outro cliente executou operações sobre o objeto X.

- A. Viola a ordem sequencial do cliente e não está de acordo com o protocolo.
- B. Pode suceder se entre o 2º e o 3º read o primário foi abaixo.
- C. Pode suceder se se perdeu a mensagem entre o primário e o secundário.
- D. Pode suceder se a operação de escrita ainda não tiver sido terminada e o cliente fizer um novo read que se executa mais rápido.

3. Suponha que no seu projeto em produção durante 1 ano, dois servidores falharam (e foram repostos corretamente em funcionamento os seus substitutos). Na situação de falta o tempo médio da troca primário-secundário observado pelos clientes foi de 10 minutos.

A. Disponibilidade = $\frac{(365 \times 24 \times 60 - 10)}{365 \times 24 \times 60}$

B. Disponibilidade = $\frac{(365 \times 24 \times 60 - 20)}{365 \times 24 \times 60}$

C. Disponibilidade = $\frac{\left(\frac{365 \times 24 \times 60}{2} - 10\right)}{365 \times 24 \times 60}$

D. MTBF dos servidores = $\frac{20}{365 \times 24 \times 60}$

4. Num sistema de réplica passiva. P é o período entre l'ím alives e T_{max} é o tempo máximo de propagação na rede. O tempo mínimo de paragem do sistema ($T_{blocking}$) é de:
- Igual ao período P .
 - Igual a $P+t_{max}$.
 - Na versão do trabalho $P+ t_{max}+ T_{modificação}$ do endereço no UDDI + T resolução do endereço no UDDI.
 - É o tempo de pôr em funcionamento outro servidor para substituir o primário que falhou.

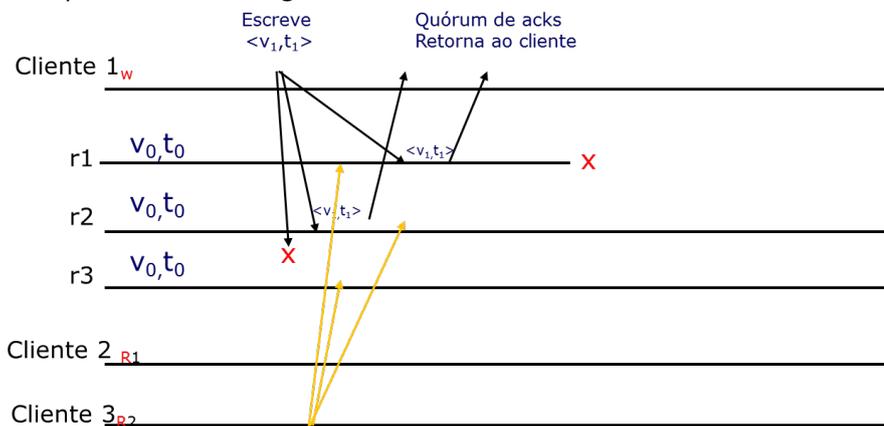
5. Num sistema de 3 réplicas que usa o protocolo *quorum consensus* (pesos idênticos, quórums de maioria), o estado das réplicas num dado instante é o seguinte (seq – sequence number, cid – client identifier):
- Réplica A: valor = 1; tag = {seq=1, client-id=1}
- Réplica B: valor = 2; tag = {seq=2, client-id=2}
- Réplica C: valor = 2; tag = {seq=2, client-id=2}

Neste instante, o front-end do cliente 5 decide escrever o valor 10. Qual a tag que esse front-end associará à escrita? Assuma que o estado acima das réplicas se mantém constante enquanto a tag da nova escrita é definida.

- {seq=2, cid=5}
- {seq=5, cid=10}
- {seq=3, cid=5}
- {seq=10, cid=5}

6. Num sistema *quorum consensus* com 5 réplicas, definir $RT=2$ e $WT=4$ (em vez de usar quórums de maioria):
- Aumenta o desempenho e disponibilidade das operações de escrita.
 - Aumenta o desempenho e disponibilidade das operações de leitura.
 - É incorrecto pois não garante a consistência do sistema.
 - Nenhuma das anteriores.

7. Considere o seguinte diagrama de um protocolo de quórums de maioria com 3 réplicas ativas. As setas representam mensagens enviadas.



- Perante o esquema o Cliente 3 vai ler V_0 porque dois dos servidores têm este valor.
- O cliente vai sempre ler V_1 porque tem o contador mais elevado.
- Se R_2 responde o resultado é V_1 .
- O diagrama não está correto porque se a mensagem de escrita não chegou a R_3 então este tem de ficar em falha silenciosa.

8. Considere o seguinte programa de uma transação distribuída, em que as contas de destino estão alojadas em servidores remotos. O protocolo é o 2PC.

```
1 boolean makeTransfers(List<TransferInfo> transfers, Account sourceAccount) {
2     Object tx = beginTransaction();
3     for (TransferInfo t : transfers) {
4         sourceAccount.transferFrom(t.targetAccount, t.amount, tx);
5     }
6     return closeTransaction(tx);
7 }
```

- A. A votação do 2PC é feita na linha 2.
- B. A votação do 2PC é feita na linha 6.
- C. A votação do 2PC é feita gradualmente, cada vez que a linha 4 se executa.
- D. No 2PC não há votação.

9. O 2-phase commit baseia-se no uso de *timeouts*.

- A. Consequentemente, o protocolo só pode ser usado em sistemas síncronos.
- B. Consequentemente, o protocolo exige relógios sincronizados.
- C. Apesar de usar *timeouts*, o protocolo pode ser usado em sistemas assíncronos.
- D. Nenhuma das anteriores.

10. No 2-phase commit, o Coordenador recebeu voto NÃO de um dos participantes.

- A. Escusa de esperar por outros votos; pode enviar imediatamente a ordem de doAbort a todos os participantes.
- B. Escusa de esperar por outros votos; pode enviar imediatamente a ordem de doCommit a todos os participantes.
- C. Espera pelos votos dos participantes em falta e só depois envia doAbort a todos.
- D. Espera pelos votos dos participantes em falta e só depois envia doCommit a todos.

1	2	3	4	5	6	7	8	9	10	Total
2	2	2	2	2	2	2	2	2	2	20