



Licenciatura / Mestrado Integrado em Engenharia Informática

Sistemas Distribuídos – 2º teste, 29 de Maio de 2015

1º Semestre, 2014/2015

NOTAS: Leia com atenção cada questão antes de responder. **O teste é sem consulta e tem a duração de 1h30min.** O teste contém **5** páginas.

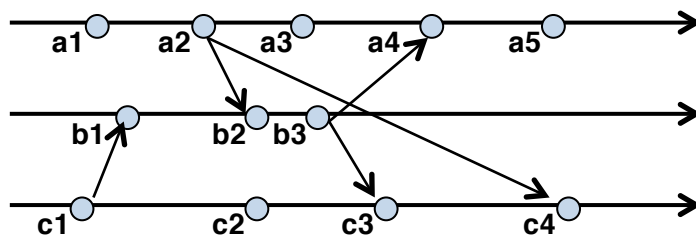
NOME: _____ **NÚMERO:** _____

1) Para cada pergunta, assinale como **V[erdadeira]** ou **F[alsa]** cada uma das afirmações. **As respostas erradas descontam.**

- ___ No Java RMI, o carregamento dinâmico do código é importante apenas para estender a funcionalidade do cliente (e não para permitir a um cliente estender a funcionalidade do servidor).
- ___ No Java RMI, a máquina virtual em que existe uma referência para um servidor comunica periodicamente com o servidor para indicar a existência desta referência, evitando o servidor de ser recolhido (garbage collected).
- ___ O WSDL inclui a especificação das mensagens enviadas entre os clientes e os servidores.
- ___ O WSDL é tipicamente gerado automaticamente a partir da interface/código do servidor através de ferramentas automáticas.
- ___ Os Web Sockets podem ser usados para permitir a um servidor notificar os clientes de novos eventos.
- ___ Uma grande vantagem da representação JSON é que pode ser manipulada diretamente por cliente JavaScript a correr num browser.
- ___ No sistema CIFS, quando um cliente abre um ficheiro para leitura/escrita é necessário obter imediatamente um op-lock exclusivo.
- ___ No caching do sistema NFS, o cliente pode aceder a informação desatualizada mesmo sem que existam falhas de comunicação.
- ___ No caching do sistema CIFS, o cliente pode aceder a informação desatualizada mesmo sem que existam falhas de comunicação.
- ___ O sistema de caching NFS não é muito eficiente para fazer cache dos ficheiros de uma base de dados quando existem acessos por múltiplos clientes.
- ___ O TCP é um sistema de comunicação que garante uma ordenação FIFO.
- ___ O TCP é um sistema de comunicação síncrono.
- ___ Um sistema de comunicação anycast é interessante para executar operações de leitura num sistema replicado.
- ___ Num sistema de multicast fiável (sem uniform agreement) é possível uma mensagem ser entregue a apenas um processo do grupo.

2) Considere um sistema distribuído com três processos, em que ocorrem os eventos assinalados a1, a2, ... As setas indicam o envio de uma mensagem.

a) Neste contexto, assinale como **V[erdadeira]** ou **F[alsa]** cada uma das afirmações. **As respostas erradas descontam.**



___ c2 aconteceu antes de a5.

___ a1 aconteceu antes de a3.

___ a1 aconteceu antes de c3.

b) Suponha que pretende identificar os eventos com relógios lógicos de Lamport. Indique o valor para cada um dos seguintes eventos, sabendo que o primeiro evento de cada processo será identificado com o relógio 1 e que cada relógio será incrementado sempre pelo menor valor possível.

a2: _____ a3: _____ b3: _____ c3: _____ c4: _____

c) Se para os eventos e1 e e2 tivermos as suas histórias causais, é possível verificar se e1 aconteceu antes de e2? Explique como ou porque não.

Sim, porque... / Não, porque...

- 3) O Last.FM fornece um serviço de informação sobre música, a qual pode ser acedida fazendo pedidos HTTP. Por exemplo, para obter informação sobre um álbum de música, o cliente pode enviar um pedido HTTP GET para o seguinte URL:

`http://ws.audioscrobbler.com/2.0/?method=album.getinfo&api_key=4cf23dfac5104&artist=Bauhaus&album=Mask&format=json`

- a) Considera que este URL respeita fielmente os princípios do desenho dos sistemas REST? Justifique - em caso negativo, a sua resposta, além da justificação, deve apresentar uma alternativa de URL para o método indicado.

Sim, porque... / Não, porque...

- b) Em termos de desempenho, que problemas é que a solução existente coloca, considerando que a API será usada por múltiplas aplicações e que os parâmetros do URL podem ser enviados por diferentes ordens.

- 4) Considere o protocolo primário-secundário estudado nas aulas. NOTA: as respostas devem ser relativa à implementação correta e completa do protocolo e não à sua implementação no trabalho prática.
- a) Ao processar um pedido, o primário deve esperar que os secundários confirmem a recepção/execução da operação antes de a enviarem ao cliente? Justifique.

Sim, porque... / Não, porque...

- b) Suponha que tinha ao seu dispor um sistema de *message-queue* que garantia a persistência das mensagens enviadas. Explique como poderia implementar o protocolo primário-secundário recorrendo a esse sistema.

OPERAÇÃO DE ESCRITA:

PRIMÁRIO:

SECUNDÁRIO:

OPERAÇÃO DE LEITURA:

- 5) Considere uma aplicação que permita a múltiplos utilizadores modificarem concorrentemente um mesmo desenho usando as seguintes operações:
- `create( id, props)` - cria objeto (e.g. círculo) com identificador único `id` com as propriedade `props`
 - `modify( id, props)` - modifica propriedades (e.g. cor, dimensão) do objeto `id`
 - `destroy( id)` - destrói objeto com identificador único `id`

Suponha que cada cliente mantém uma cópia do estado partilhado do desenho e que se pretende que o estado convirja para o mesmo estado nas réplicas de todos os clientes. Se pretendesse usar um sistema de comunicação multi-ponto para implementar o sistema, indique justificadamente que propriedade mínimas deveria exigir para a execução de cada operação.

create:

modify:

destroy:

- 6) No sistema NFS, explique como é que um cliente detecta que um ficheiro que tem em cache foi modificado no servidor.