

Disciplina DCE540 - Computação Paralela e Distribuída	Método de realização Presencial	Data da prova 01/02/2023 às 14h00
Professor Iago Augusto de Carvalho (iago.carvalho@unifal-mg.edu.br)		

Prova 03

Exercício 1 (15%)

Apresente as definições de

- a) Réplicas inicializadas pelos clientes
- b) Réplicas inicializadas pelo servidor (ou pelo sistema distribuído)
- c) Réplicas permanentes

Exercício 2 (20 %)

Sobre a confiabilidade em sistemas distribuídos. Quais são as duas métricas utilizadas para computar a confiabilidade de um sistema distribuído? Apresente a definição destas duas métricas e explique como você as utilizaria para calcular a confiabilidade de um sistema distribuído.

Exercício 3 (15%)

Falhas podem ser transientes, intermitentes ou permanentes. Apresente a definição dos três tipos de falhas e aponte qual é a mais simples e qual é a mais difícil de ser identificada e tratada.

Exercício 4 (15 %)

Qual é a utilidade da redundância na tolerância a falhas? Tome como base o sistema sistema distribuído por trás de um grande aplicativo, como o iFood. Como você implementaria redundância de informação e redundância física neste sistema?

Exercício 5 (15 %)

Quais são as diferenças entre erro, falha e falta? Escolha um componente de um sistema distribuído (um computador, um banco de dados, uma impressora, um software, um meio de comunicação, um serviço em nuvem, dentre outros) e apresente um erro, uma falha e uma falta para este componente.

Exercício 6 (20%)

Em relação a tolerância a falhas, apresente as definições de

- a) Confiabilidade
- b) Manutenibilidade
- c) Disponibilidade
- d) Segurança

Gabarito

Exercício 1

- a) Estas são cópias *cache* que cada cliente pode manter em seu próprio dispositivo (ou em algum dispositivo de rede próximo a ele, e.g., um roteador LAN). Este tipo de réplica é muito útil quando um cliente faz um grande número de requisições a um mesmo conjunto de dados para diminuir o tempo de acesso e leitura. Estas réplicas, assim como as inicializadas pelo servidor, são temporárias. Além disso, a atualização dos dados de uma réplica inicializada pelo cliente é do próprio cliente e não do sistema distribuído.
- b) Em sua maioria, são réplicas temporárias utilizadas para realizar o balanceamento de carga. Estas réplicas são criadas pelo sistema distribuído quando existe um número excessivo de acesso a um conjunto de dados (por exemplo, durante a realização de eventos promocionais). Além disso, estas réplicas costumam ser somente leitura (*read-only*) e seu único objetivo é prover um serviço a um grande número de clientes
- c) Réplicas permanentes são aquelas criadas junto com o próprio sistema distribuído. Normalmente elas são utilizadas como cópias de segurança dos dados e também visam oferecer um pequeno balanceamento de carga das ações. Estas réplicas permanentes costumam ser poucas e, muitas vezes, servem como espelho dos dados do sistema distribuído.

Exercício 2

A primeira métrica é o tempo médio para falha (T_f), que mede de quanto em quanto tempo uma falha ocorre. Já a segunda métrica é o tempo médio para recuperação (T_r), que mede quanto tempo o sistema leva para se recuperar de determinada falha.

Pode-se calcular a confiabilidade de um sistema distribuído utilizando a fórmula

$$\text{Confiabilidade} = \frac{T_f}{T_f + T_r}$$

Exercício 3

Transiente: É aquela falha que ocorre uma única vez, muitas vezes devido a um evento físico excepcional. Mesmo que a ação que levou a falha seja repetida, ela não ocorrerá novamente. **Intermitente:** São falhas ocasionais, que ocorrem de vez em quando. Não tem um motivo definido, podendo ser ocasionadas por *bugs* em códigos, eventos físicos ou entradas não previstas no sistema distribuído. **Permanentes:** São falhas perpétuas, que não podem ser recuperadas sem a troca ou substituição do componente defeituoso.

O tipo de falha mais difícil de ser identificada e tratada e recuperada é a intermitente, pois ela depende da observação do sistema distribuído no exato momento de ocorrência da falha, sendo este momento ocasional.

Exercício 4

Redundância é útil para fazer com que as falhas de um sistema distribuído (ou de um componente) sejam transparentes a seus usuários. Em um grande aplicativo, como o iFood, poderíamos implementar redundância de informação ao prover códigos de verificação de erros em todas as requisições e respostas com origem ou destino, seja na compra de alimentos e bebidas ou no sistema de login. Já a redundância física poderia ser atendida ao criarmos cópias físicas dos servidores que hospedam o *backend* do aplicativo iFood, junto de um *middleware* inteligente que fosse capaz de tratar as falhas destes diferentes servidores. Para um sistema do tamanho do iFood, também seria interessante implementarmos redundância física utilizando técnicas de DevOps e hospedagem em nuvem.

Exercício 5

Uma **falha** é quando um componente deixa de ser útil ou funcional, isto é, quando ele não consegue mais cumprir suas tarefas. Já um **erro** refere-se ao estado (ou parte) do componente que deu origem a falha. Já uma **falta** é o que levou ao erro.

Vamos tomar como exemplo de componente uma impressora em rede, conectada a um sistema distribuído utilizando um cabo de rede. Pode-se dizer que a impressora falhou quando ela deixa de responder apropriadamente as requisições a ela enviadas. Por exemplo, quando usuários enviam arquivos para impressão e ela não consegue mais imprimir. Um possível erro referente a esta falha pode estar localizado nos cartuchos de tinta da impressora, sendo que a falta relacionada é o esgotamento dos cartuchos.

Exercício 6

- a) Diz que o sistema (ou componente) tem que funcionar continuamente sem interrupção do serviço. Confiabilidade faz referência ao espaço de tempo que o sistema (ou componente) opera sem apresentar nenhuma falha.
- b) Refere-se ao tempo de recuperação de um sistema (ou componente) após uma falha e a facilidade de recuperação do mesmo após a falha.
- c) Diz que o sistema (ou componente) tem que estar pronto para uso sempre que for requisitado por um usuário. Ou seja, ele tem que estar pronto para uso imediato sempre que for requisitado.
- d) Diz que, no caso de uma falha, nenhum evento catastrófico pode ocorrer. Um sistema seguro é aquele cujas falhas não resultem em eventos extremamente graves a seus usuários ou a outras pessoas.