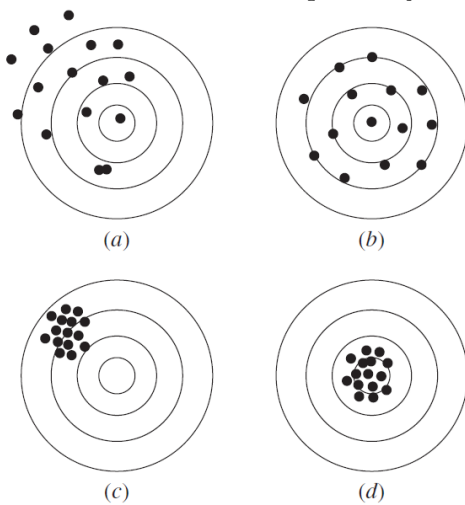


Disciplina DCE540 - Computação Paralela e Distribuída	Método de realização Presencial	Data da prova 07/12/2022 às 14h00
Professor Iago Augusto de Carvalho (iago.carvalho@unifal-mg.edu.br)		

Prova 02

Exercício 1 (15 %)

Quando estamos trabalhando com algoritmos de sincronização de *clock*, devemos garantir uma acurácia e uma precisão mínima para todos os *clocks* de um sistema distribuído. Observe a imagem abaixo com quatro diferentes alvos e considere que o objetivo é acertar no centro do alvo. Neste contexto, responda e justifique:



- Qual alvo apresenta resultados com uma alta acurácia, mas baixa precisão?
- Qual alvo apresenta resultados com uma alta precisão, mas uma baixa acurácia?

Exercício 2 (10 %)

Sobre sincronização de clocks, assinale **V** para verdadeiro e **F** para falso e justifique

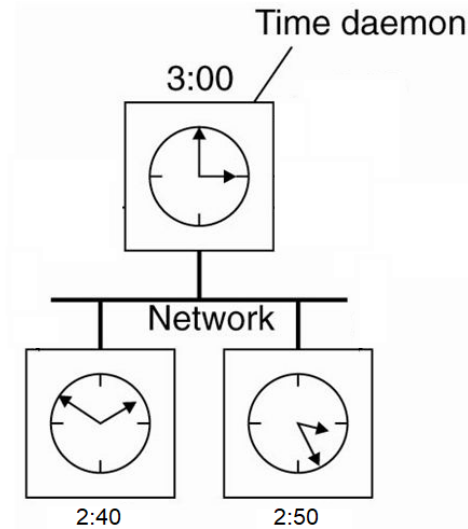
- Em sistemas não-distribuídos, o clock é uma ótima forma de realizar sincronização, como o make em sistemas UNIX.
- No algoritmo de Berkley, um servidor atualiza frequentemente o clock dos outros computadores, e essa sincronização é sempre atualizando os computadores para a hora atual do servidor.
- Utilizando o clock lógico, nem todos os computadores precisam estar com os clocks sincronizados.
- O UTC (Universal Coordinated Time) pode ser considerado como um clock global.

Exercício 3 (15%)

Qual é a principal ideia por trás da utilização do Algoritmo de Berkley? Explique, com suas próprias palavras, as principais vantagens e desvantagens da utilização deste método de sincronização.

Exercício 4 (15%)

Ainda sobre o Algoritmo de Berkley. Veja a imagem abaixo e mostre o funcionamento do algoritmo, dizendo qual será o tempo atualizado em todas as máquinas



Exercício 5 (15%)

Algoritmos de exclusão mútua coordenam o acesso a recursos compartilhados. Estes algoritmos devem garantir que não ocorram *deadlocks*, que não exista *starvation* e que o acesso aos recursos compartilhados sejam realizados com equidade e justiça. Explique, com suas próprias palavras, os conceitos de

- a) *Deadlock*
- b) *Starvation*
- c) *Equidade e justiça*

Exercício 6 (30%)

Algoritmos de eleições são utilizados para eleger processos coordenadores em sistemas distribuídos, isto é, aqueles processos que são responsáveis por gerenciar o acesso aos recursos compartilhados. Apresente e descreva as vantagens e desvantagens do algoritmo.

- a) Algoritmo de Bully
- b) Algoritmo do anel (token-ring)
- c) Algoritmo de eleições para redes de larga escala

Gabarito

Exercício 1

- a) Acurácia refere-se a diferença entre o *clock* de quaisquer dois dispositivos pertencentes a um mesmo sistema distribuído. Neste sentido, pode-se entender que os alvos com alta acurácia são o (c) e o (d). Entretanto, a precisão de (d) é alta, enquanto a precisão de (c) é baixa. Assim, a resposta correta é a alternativa (c).
- b) Precisão refere-se a diferença entre o *clock* de um dispositivo de um sistema distribuído e um *clock* de acreditação global, como o UTC. Neste sentido, devemos procurar os alvos com as marcas mais próximas do centro. Observa-se que tanto os alvos (b) como (d) tem alta precisão. Entretanto, a acurácia de (d) é alta, enquanto a de (b) é baixa. Assim, a resposta correta é a alternativa (b)

Exercício 2

- a) **Verdadeiro.** Como o sistema é não-distribuído e existe em um único dispositivo, o clock é único. Assim, ele é uma ótima maneira de sincronizar serviços e softwares.
- b) **Falso.** A atualização é realizada levando em consideração a média dos tempos e não a hora atual do servidor
- c) **Verdadeiro.** Neste esquema só é preciso sincronizar os processos (ou computadores) que utilizam os mesmos recursos compartilhados
- d) **Verdadeiro.** Por definição, o UTC é um clock global

Exercício 3

Este algoritmo sincroniza o clock de todos os dispositivos levando em consideração a média dos tempos de todos eles. A principal vantagem é que ele não depende de um clock global para ser utilizado, podendo ser implementado e executado sem realizar requisições a servidores externos. Como desvantagem, pode-se apontar a falta de sincronia dos clocks do sistema distribuído para o UTC.

Exercício 4

São três dispositivos, sendo um marcando as 2h40, o segundo marcando as 2h50 e o último (o coordenador) marcando as 3h00.

Em um primeiro passo, o coordenador envia uma requisição para cada máquina do sistema distribuído pedindo o seu clock atual. Em um segundo passo, as máquinas, então, enviam seus clocks ao coordenador. Em um terceiro passo, o coordenador computa a média dos clocks que, então, é enviado de volta as outras máquinas em um quarto passo. A hora final do sistema (atualizado) é essa média dos clocks.

No caso desta figura, a hora final seria igual a

$$\frac{(3h00 - 3h00) + (2h40 - 3h00) + (2h50 - 3h00)}{3} = \frac{0 - 0h10 - 0h20}{3} = \frac{-0h30}{3} = -0h10$$

$$3h00 - 0h10 =$$

Exercício 5

- a) **Deadlock:** Quando dois ou mais processos ficam impedidos de continuar suas execuções - ou seja, ficam bloqueados, esperando uns pelos outros. Isto ocorre quando um processo A está acessando um recurso compartilhado X e um segundo processo B está acessando um recurso compartilhado Y. Entretanto, o processo A quer acessar Y e o processo B quer acessar X. Nenhum dos dois processos cedem e, assim, ocorre o deadlock

- b) **Starvation:** Ocorre quando um processo tenta acessar um recurso compartilhado, mas nunca consegue. Normalmente, este recurso compartilhado está bloqueado por outro processo. Desta forma, dizemos que o processo está *passando fome*, ou seja, não consegue acessar o recurso que quer
- c) *Equidade e justiça:* Refere-se ao fato de que todos os processos tem que ter as mesmas chances de acessar um recurso compartilhado, sem distinções entre eles.

Exercício 6

- a) **Algoritmo de bully:** Neste algoritmo, quando algum processo percebe que o coordenador falhou, todos os processos (ativos e inativos) recebem um identificador numérico e é dado início as eleições. O processo de eleições consiste na troca de mensagens entre os diversos processos do sistema distribuído. Estas mensagens são sempre enviadas dos processos com menor valor de identificador para aqueles com maior valor. Assim, temos duas opções:

- (a) Se um processo recebe uma mensagem e ele está ativo, então ele responde com um *OK* e retira o emissor da mensagem das eleições.
- (b) Se o processo está inativo e não consegue responder a mensagem, nada ocorre.

Este processo ocorre de forma iterativa até que um processo de identificador alto não receba mensagens de nenhum outro com valor de identificador acima dele. Logo após, ele envia uma mensagem para todos os outros processos do sistema informando que ele é o novo processo coordenador. **Vantagens:** Um único coordenador é eleito em todas as situações. **Desvantagens:** É necessário realizar um grande número de troca de mensagens. Pode ser um pouco lento devido a demora na resposta de processos inativos

- b) **Algoritmo do anel:** Neste algoritmo, assume-se que os processos são organizados na forma de um círculo (ou anel) e que cada processo conhece quem é seu sucessor. Quando é identificado que processo coordenador falhou, faz-se necessário eleger um novo coordenador. A eleição é inicializado pelo(s) processo(s) que perceber(am) que o coordenador falhou. A eleição consiste em circular uma mensagem na rede duas vezes: a primeira para verificar quais processos estão ativos no momento e a segunda para informar a todos os processos qual é o novo coordenador, que será aquele que iniciou a eleição. **Vantagens:** Simples implementação. **Desvantagens:** Um número muito grande de mensagens deve ser trocada, o que pode sobrecarregar a rede; Não funciona para sistemas distribuídos muito grandes; Caso dois ou mais processos iniciem as eleições, é necessário um mecanismo adicional para decidir qual deles será o coordenador.
- c) **Eleições em larga escala:** Este algoritmo tenta eleger múltiplos coordenadores de tal forma que eles estejam uniformemente espalhados pela rede. **Vantagens:** Pode ser utilizado em redes de larga escala; Consegue eleger múltiplos coordenadores; Muito eficiente em redes dinâmicas, onde existe a entrada e saída frequente de novos usuários ou componentes **Desvantagens:** Em redes pequenas, múltiplos coordenadores pode dificultar o acesso a recursos compartilhados