

UNIVERSIDADE FEDERAL FLUMINENSE
INSTITUTO DE COMPUTAÇÃO
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

Sistemas Distribuídos 2019.1

Profa.: Simone Martins

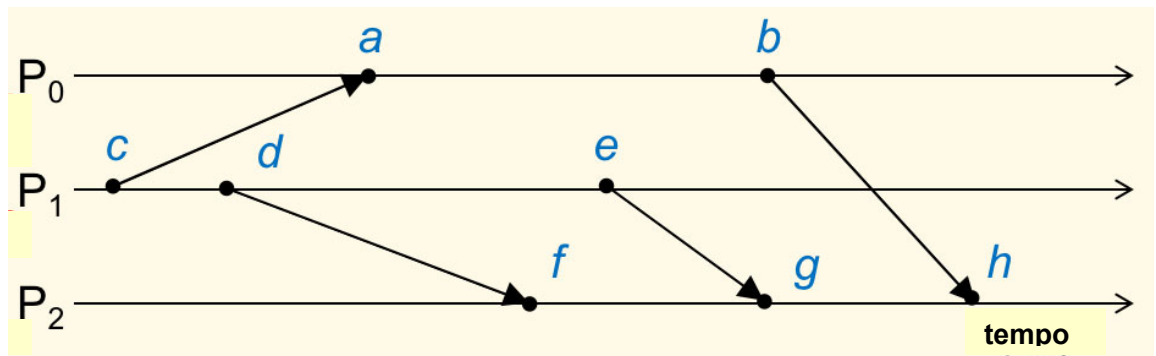
GABARITO P2

Nome:

1. (1,4) Considere que, em um sistema que usa a sincronização de relógio de Berkeley, o coordenador tem a hora igual a 2:20:00 e inicia uma fase de sincronização de relógio. Dois sistemas escravos A e B que sincronizam com o coordenador tem as horas 2:20:04 e 2:20:05. Calcule a hora para a qual os três irão acertar seus relógios, após a fase de sincronização.

Resposta: $(2:20:00 + 2:20:04 + 2:20:05) / 3 = 2:20:03$

2. (1,6) Indique, **explicando**, a marca de hora de Lamport que terão os eventos a,b,c,d,e,f,g, e h. Assuma que o valor inicial da marca de hora de Lamport tem o valor igual a 0 em cada processo P_0 , P_1 e P_2 .



Resposta: $a=2$ (maior entre $(0,1) + 1$, $b=3$ (incrementa de 1 em ocorrência de evento), $c=1$, $d=2$ (incrementa de 1 em ocorrência de evento), $e=3$ (incrementa de 1 em ocorrência de evento), $f=3$ maior entre $(0,2)+1$, $g=4$ maior entre $(3,3)+1$, $h=5$ maior entre $(3,4)+1=5$

3. (3,2) Para a figura abaixo, **explique** se a sequência de eventos pode ser permitida em um sistema que deve ter consistência sequencial e em um sistema que deve ter consistência causal. Para o(s) caso(s) em que a sequência não seja permitida, indique, **explicando**, uma outra ordem dos mesmos eventos que seria consistente.

P1:	W(x)a	W(x)c		
P2:	R(x)a	W(x)b		
P3:	R(x)a		R(x)c	R(x)b
P4:	R(x)a		R(x)b	R(x)c

Resposta: Caso esta sequência ocorra, temos um sistema com consistência causal porque todas as leituras de x com o valor a,b e c são realizadas depois da escrita de x com o valor a, b e c respectivamente. Não temos um sistema com consistência sequencial, porque não existe uma única ordem de execução das operações de escrita e leitura nos 4 processos. Por exemplo, P3 pode ler o valor b para x, caso a operação de escrita do valor b em x seja recebida depois da operação de escrita do valor c em x. Mas essa ordem não explica a leitura

do valor b de x e depois c de x no Processo 4. Caso Rx (b) acontecesse antes de Rx(c) em P3 ou Rx (c) acontecesse antes de Rx(b) em P3 teríamos uma consistência sequencial

4. (2,4) Considere um sistema cliente-servidor em que um cliente envia um documento para ser impresso em um servidor. O protocolo a ser seguido por ambos é descrito a seguir. O cliente envia um pedido de impressão para o servidor que envia uma confirmação para o cliente, caso possa realizar a impressão. O servidor após enviar a confirmação, imprime o documento e envia uma mensagem de conclusão para o cliente. Caso ocorra uma queda do servidor, o cliente será avisado que houve a queda quando o servidor se recuperar. Um cliente, ao ser notificado da recuperação do servidor, pode utilizar as seguintes estratégias de retransmissão da solicitação do pedido de impressão: (i) sempre; (ii) nunca; (iii) somente quando tiver recebido confirmação antes da queda e (iv) somente se não tiver recebido confirmação antes da queda.

Para cada um dos cenários de queda do servidor abaixo, **explique se** o documento será impresso uma vez (OK), não será impresso (NOK) ou será impresso duas vezes (DUP), para cada uma das quatro estratégias de retransmissão.

4.1. PMC: o servidor imprime, envia a mensagem de conclusão e sofre uma queda

Resposta: (i) DUP, porque o servidor imprimiu e o cliente vai solicitar novamente
(ii) OK, o servidor imprimiu e não vai imprimir novamente porque cliente não vai pedir
(iii) DUP, porque o servidor imprimiu e o cliente vai solicitar novamente
(iv) OK, o servidor imprimiu e não vai imprimir novamente porque cliente não vai pedir

4.2. PC(M): o servidor imprime e sofre uma queda, portanto a mensagem de conclusão não é enviada

Resposta: (i) DUP, porque o servidor imprimiu e o cliente vai solicitar novamente
(ii) OK, o servidor imprimiu e não vai imprimir novamente porque cliente não vai pedir
(iii) DUP, porque o servidor imprimiu e o cliente vai solicitar novamente
(iv) OK, o servidor imprimiu e não vai imprimir novamente porque cliente não vai pedir

4.3. C(PM): o servidor sofre a queda antes de imprimir e enviar a mensagem de conclusão

Resposta: (i) OK, porque o servidor não imprimiu e o cliente vai solicitar novamente
(ii) NOK, o servidor não imprimiu e não vai imprimir novamente porque cliente não vai pedir
(iii) OK, porque o servidor não imprimiu e o cliente vai solicitar novamente
(iv)) NOK, o servidor não imprimiu e não vai imprimir novamente porque cliente não vai pedir

5. (1,4) Explique como podem ser utilizados pontos de verificação para recuperar um sistema distribuído após a falha de um dos processos e mostre um exemplo do efeito dominó que pode ocorrer quando estes pontos de verificação são independentes.

Resposta: Cada processo salva seu estado periodicamente em um armazenamento estável disponível localmente. A recuperação após uma falha de processo ou de sistema requer a construção de um estado global consistente com base nesses estados locais. A melhor alternativa é recuperar a fotografia mais recente, denominada linha de recuperação que corresponde ao mais recente conjunto consistente de pontos de verificação. Um conjunto consistente é composto por pontos de verificação nos quais os recebimentos de mensagens por um processo estão no ponto de verificação do processo e os envios das mesmas mensagens estão no ponto de verificação do processo que enviou as mensagens.

Exemplo de efeito dominó.

Neste caso não se consegue uma linha de recuperação consistente com a combinação dos pontos de verificação dos dois processos, então se tem que voltar o estado inicial.

