



Disciplina: Sistemas Distribuídos
Professora: Ana Cristina Barreiras Kochem Vendramin

Avaliação (valor 2,0)
**Sincronização Interna de Relógios Físicos,
Eleição, Tolerância a Falhas e Segurança.**

Siga as instruções abaixo para desenvolver e testar um sistema que permita a sincronização de relógios físicos através do algoritmo de Berkeley:

1. Considere um conjunto mínimo inicial de quatro processos que precisam sincronizar internamente seus relógios. Um desses processos atuará como o servidor do tempo e será chamado de mestre. Os outros processos serão os escravos.
2. Utilize a comunicação em grupo (*multicast*) para que os quatro processos se conheçam, troquem suas chaves públicas e elejam um para atuar como o mestre. Os processos devem registrar localmente o mestre atual (valor 0,3).
3. O sistema deve ser tolerante a falhas do processo mestre. Utilize a comunicação em grupo para o mestre anunciar a sua disponibilidade a cada Δt_1 intervalos de tempo. O não recebimento do anúncio dentro de Δt_1 indicará falha no mestre. Se o processo mestre falhar, um dos escravos deverá ser eleito para assumir o seu lugar (valor 0,5).
4. A cada Δt_2 intervalos de tempo, o mestre perguntará o horário marcado em cada escravo. Os escravos retornam seus valores de relógios (t). Ao receber as respostas, o mestre calcula os tempos de viagem de ida e volta (RTT – *Round Trip Time*) de cada pacote enviado para cada escravo. O mestre estima o horário de cada escravo empregando a técnica de Cristian ($t + RTT/2$). Com o tempo estimado, o mestre faz a média dos valores obtidos (**incluindo a leitura de seu próprio relógio**). Ao invés de enviar para os escravos o horário corrente atualizado, o que introduziria mais incerteza devido ao tempo de transmissão do pacote, o mestre envia um ajuste para que cada escravo adiante ou atrase seu relógio. Ao receber o ajuste do mestre, os processos escravos devem

atualizar seus relógios físicos. Toda comunicação deve ser unicast (valor 0,7).

5. O sistema deve evitar que um processo mal-intencionado assuma o papel de mestre e introduza valores incorretos no sistema. Para isso, a cada envio de ajuste de relógio, é necessário empregar chaves assimétricas (chave pública e privada) para assegurar a cada escravo que o mestre eleito é o processo que está enviando o ajuste em questão (valor 0,5).
6. É obrigatório documentar todo o código e a equipe é de dois programadores.