

1) Tanto o algoritmo de exclusão mútua de Lamport quanto o algoritmo de Ricart e Agrawala são projetados para garantir exclusão mútua em sistemas distribuídos. No entanto, a maneira como cada um lida com a questão de justiça em relação ao uso da seção crítica pode ser diferente.

O algoritmo de exclusão mútua de Lamport é um algoritmo baseado em relógio lógico que usa um pedido de entrada para obter acesso exclusivo a uma seção crítica. Se um processo solicita acesso a uma seção crítica e outro processo já está usando a seção crítica, o processo que solicita deve aguardar até que sua solicitação seja atendida. Isso garante que o acesso à seção crítica seja concedido de forma justa a cada processo na ordem em que as solicitações são recebidas.

Por outro lado, o algoritmo de Ricart e Agrawala é um algoritmo baseado em troca de mensagens que usa uma fila de solicitações para determinar a ordem de acesso à seção crítica. Cada processo mantém uma fila de solicitações pendentes de outros processos. Quando um processo solicita acesso à seção crítica, ele envia uma mensagem para cada outro processo para solicitar sua permissão. Cada processo responde à solicitação, adicionando-a à sua própria fila de solicitações pendentes. Quando a solicitação do processo é a próxima na fila, ele tem permissão para entrar na seção crítica. Isso garante que a seção crítica seja usada de forma justa, com cada processo tendo acesso à seção crítica em ordem de chegada à fila.

No entanto, em ambos os algoritmos, se um processo usar a seção crítica várias vezes antes de permitir que outro processo a use, isso não seria considerado uma violação da justiça. Esses algoritmos são projetados para garantir exclusão mútua, não para garantir igualdade no uso da seção crítica.

2) DMUTEX (Distributed Mutual Exclusion) é um algoritmo proposto por Toshimitsu Maekawa em 1985 para garantir exclusão mútua em sistemas distribuídos com N processos. DMUTEX permite que qualquer processo solicite acesso a uma seção crítica e garante que apenas um processo tenha acesso à seção crítica a qualquer momento.

A ideia do DMUTEX é baseada em particionamento de grupos. O conjunto de processos é dividido em N/k grupos, onde k é um parâmetro que determina o tamanho dos grupos. Cada grupo tem um processo representante que é responsável por gerenciar as solicitações de acesso à seção crítica para todos os processos no grupo. O algoritmo usa um sistema de votação para determinar qual processo terá acesso à seção crítica.

O algoritmo DMUTEX funciona da seguinte maneira:

1. Cada processo envia uma mensagem de solicitação para seu representante de grupo local.
2. O representante do grupo local coleta todas as mensagens de solicitação de seus processos membros e as envia para os representantes dos outros grupos.
3. Cada representante do grupo recebe as mensagens de solicitação de todos os grupos, realiza uma votação para determinar qual processo terá acesso à seção crítica e envia o resultado da votação para todos os representantes de grupo.
4. O representante de grupo local recebe o resultado da votação e notifica o processo vencedor da votação que ele pode entrar na seção crítica.

Por exemplo, se temos um sistema com 6 processos divididos em 3 grupos ($k = 2$), o processo P1 pode solicitar acesso à seção crítica enviando uma mensagem para o representante do grupo G1. O representante do grupo G1 coleta a mensagem de solicitação de P1 e a mensagem de solicitação do

outro processo em seu grupo, P2, e envia ambas as mensagens de solicitação para os outros dois representantes de grupo. Os representantes de grupo G2 e G3 também coletam as mensagens de solicitação de seus grupos locais e enviam uma mensagem de voto para determinar qual processo terá acesso à seção crítica. O resultado da votação é recebido por todos os representantes de grupo e o representante de grupo local notifica o processo vencedor, permitindo que ele entre na seção crítica.

Comparado com os algoritmos de exclusão mútua de Lamport e Ricart Agrawala, o DMUTEX apresenta algumas vantagens e desvantagens. O DMUTEX é escalável e eficiente, já que o número de mensagens trocadas entre os processos é proporcional ao número de grupos, em vez do número total de processos. No entanto, o DMUTEX pode ser menos eficiente do que os outros algoritmos quando o tamanho dos grupos é grande e o número de grupos é pequeno. Além disso, DMUTEX requer que todos os processos conheçam todos os outros processos do sistema, o que pode ser uma limitação em sistemas grandes e dinâmicos.