Departamento de Sistemas e Computação – FURB
Curso de Ciência da Computação
Disciplina de Sistemas Distribuídos

Algoritmos de Exclusão Mútua

Prof. Aurélio Hoppe

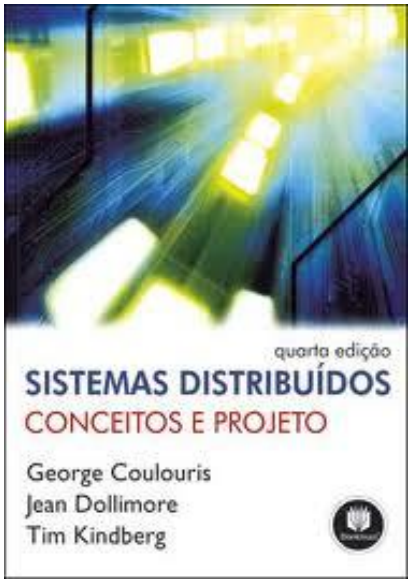
aureliof@furb.br

http://www.inf.furb.br/~aurelio/

Grupo de Processamento de Imagens,
Análise de dados, Robótica e
Simulação computacional

. . . aula de hoje

- Como realizar a obtenção da exclusividade para processos
  - Exclusão mútua
    - Algoritmo centralizado
    - Algoritmo distribuído
    - Algoritmo token ring



Exclusão mútua

- Questão fundamental em SDs:
  - Concorrência e colaboração entre vários processos
  - Processos vão precisar acessar simultaneamente os mesmos recursos
  - Para evitar que acessos concorrentes corrompam o recurso ou o tornem inconsistente acesso
  - Como garantir?
    - ter processos mutuamente exclusivos

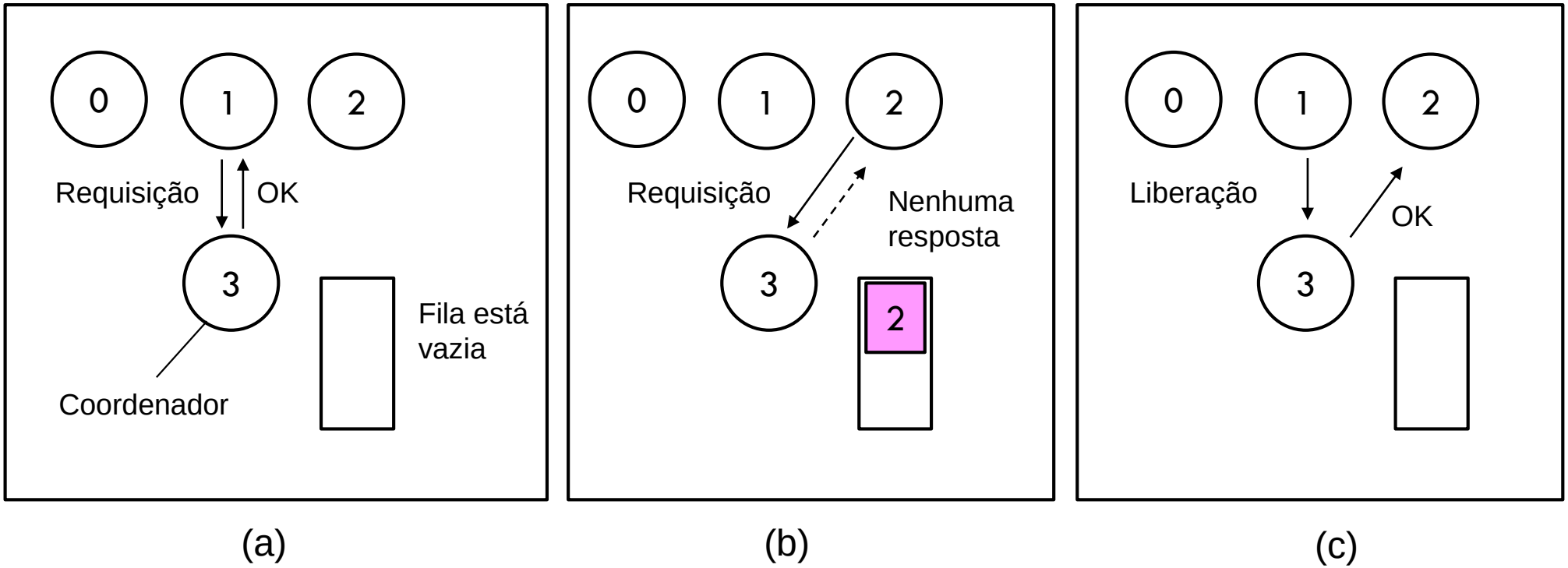
Exclusão mútua

- Duas categorias diferentes:
  - Soluções baseadas permissão:
    - Algoritmo Centralizado
    - Algoritmo Distribuído
  - Soluções baseadas token:
    - Algoritmo token ring
      - Token - Passagem de uma mensagem especial entre os processos

Algoritmo centralizado

- Coordenador (Líder):
  - sempre que um processo quiser acessar um recurso compartilhado, envia uma mensagem de requisição ao líder
  - se nenhum outro processo estiver acessando o recurso, líder devolve uma resposta, concedendo permissão
  - caso exista um processo acessando o recurso, pedido deverá ser armazenado em uma fila de espera
  - Quando processo termina de executar processo mandar mensagem de liberação para o coordenador

Algoritmo centralizado



Algoritmo centralizado

- Conclusão:
  - Garante exclusão mútua: coordenador permite o acesso de um processo por vez ao recurso
  - Fácil de implementar: requer somente três mensagens por utilização de recurso (requisição, concessão, liberação)

Algoritmo centralizado

- Problema:
  - Coordenador é um único ponto de falha, logo se ele quebrar, todo o teu sistema pode vir abaixo
  - Em um sistema grande, um único coordenador pode tornar-se o gargalo do desempenho

Algoritmo distribuído

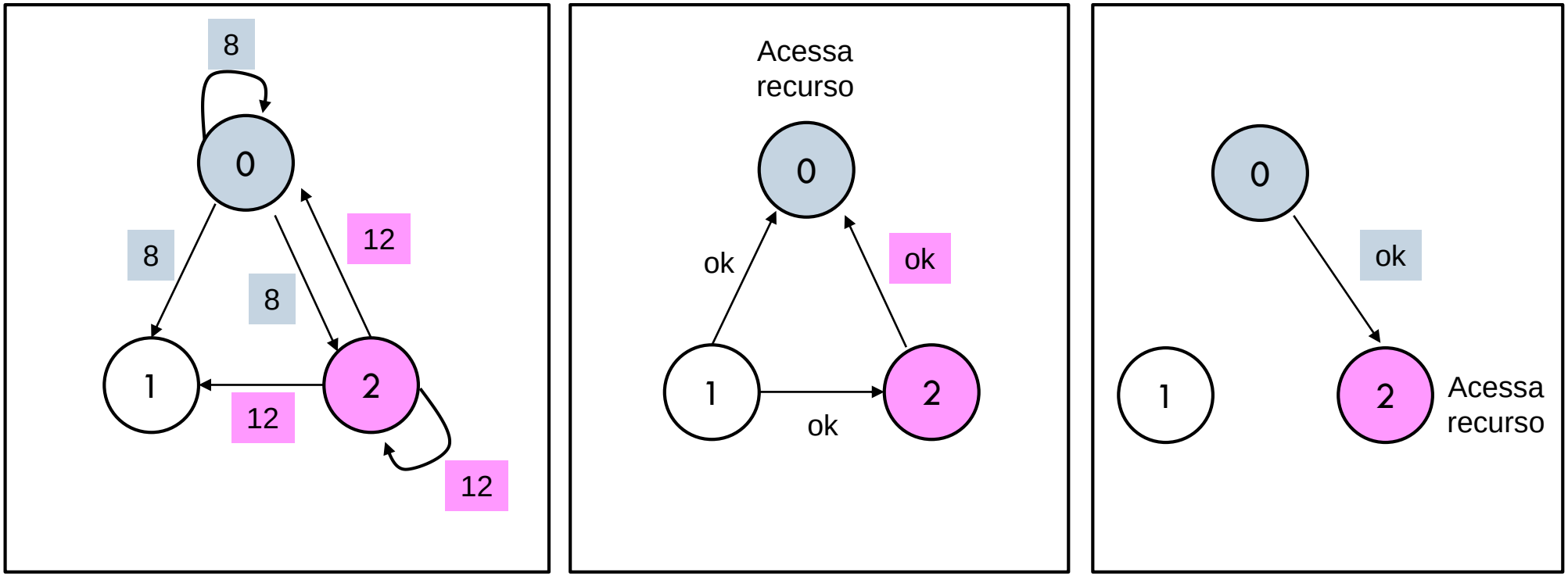
- É um algoritmo que leva em consideração a ordenação de todos os eventos do sistema

- Quando um processo quer acessar um recurso compartilhado:
  - Monta mensagem que contém o nome do recurso, seu número de processo e a hora corrente
  - Envia a mensagem a todos os outros processos
  - Premissa: mensagens não se perdem

Algoritmo distribuído

- Quando um processo recebe uma mensagem:
  - se o receptor não estiver acessando o recurso e não quiser acessá-lo: mensagem de OK
  - se o receptor já tiver acesso ao recurso, não responde e coloca a requisição em uma fila
  - se o receptor também quiser acessar o recurso, compara a marca da mensagem
    - a mensagem com marca menor vence.
    - caso a marca da mensagem que chegou seja menor, envia OK.
    - caso contrário, enfileira a requisição

Algoritmo distribuído



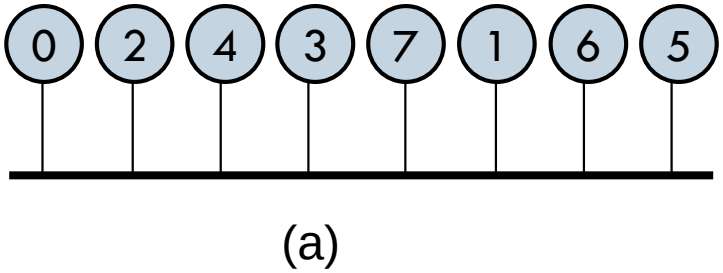
(a) (b) (c)

Algoritmo distribuído

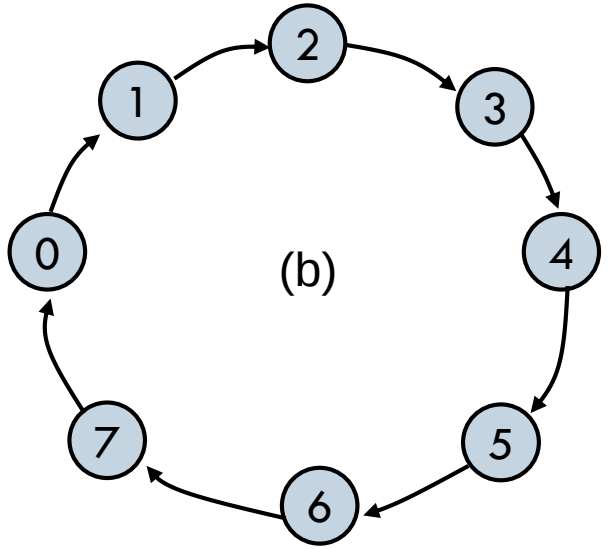
- Problema:
  - Número de mensagens trocadas (alto consumo de transmissão via rede)

Algoritmo Token Ring

- Exclusão mútua por esquemas determinísticos em um SD
  - Anel lógico
    - cada processo é designada uma posição no anel
  - Cada processo deve saber quem deve receber o token



(a)



Algoritmo Token Ring

- Quando o anel é inicializado, o processo 0 recebe o token
- O token é repassado do processo k para o k+1, através de mensagens ponto-a-ponto
- Quando um processo recebe o token, verifica se precisa acessar o recurso compartilhado
    - Acessa o recurso e repassa o token ao próximo processo
    - Caso contrário, simplesmente repassa o token

Algoritmo Token Ring

- Problema:
  - se o token se perder, deve ser gerado novamente
  - se um processo falha, este pode ser removido do grupo e o token é repassado para o próximo vizinho lógico

Revisão

- Como realizar a obtenção da exclusividade para processos
  - Exclusão mútua
    - Algoritmo centralizado
    - Algoritmo distribuído
    - Algoritmo token ring

próxima aula . . .

- Algoritmos de detecção de deadlock