

Atividade Aula 11

Aluno: *Matheus Kawamura Ferreira*

Data: 13/05/2019

1. Nos estados de processos entra o New, Running, Waiting, Terminated.
 - a. Running: seria um processo executando instruções.
 - b. Waiting: seria um processo esperando a ocorrência de algum evento.
 - c. Terminated: seria quando o processo terminasse a execução.
2. Um drive possui muitas definições, mas no sentido mais simples, um drive é um software que permite que o sistema operacional e um dispositivo comuniquem um com o outro.
3. Passo 1. Primeiro vá até o menu 'Iniciar' e acesse a seção 'Computador' ou 'Meu Computador';
Passo 2. Em seguida, clique em 'Propriedades do Sistema';
Passo 3. Na lista à esquerda, clique na opção 'Configurações avançadas do sistema'. Uma nova janela se abrirá;
Passo 4. Na barra superior da nova janela aberta, clique em 'Avançado > Desempenho > Configurações'.
Passo 5. Outra janela será aberta. Escolha 'Avançado > Alterar';
Passo 6. A janela 'Memória Virtual' se abrirá. Desmarque a opção 'Gerenciar automaticamente o tamanho do arquivo de paginação de todas as unidades' e marque 'Tamanho personalizado';
Passo 7. Em seguida, observe atentamente o espaço disponível e subtraia o espaço utilizado. Preencha as lacunas com o resultado. Depois, clique em 'Definir' e 'Ok'. Clique em 'Ok' para todas as janelas seguintes;
Passo 8: reinicie o computador para a efetivar o aumento da memória virtual.
4. Vale uma análise do cenário em que estamos e então avaliar e ponderar entre conveniência e correção. Sistemas Operacionais de propósito geral como Windows e Unix adotam essa abordagem pois em caso de deadlock o usuário terá de abandonar o processo. Não gera tantos prejuízos quando comparados com sistemas de propósito específico que são de alto risco, como por exemplo, controladores de aviões, equipamentos médicos;

Detecção e Recuperação:

- **Preempção:** retirar um recurso de algum outro processo. Nesse método, cabe uma análise da natureza do recurso;
- **Restauração de Estado:** armazenar pontos de verificação de processos periodicamente para que esse estado salvo sirva para restaurar o processo se este é encontrado em estado de deadlock;
- **Eliminação de Processos:** forma mais grosseira, mas mais simples de quebrar um deadlock é eliminar um dos processos no ciclo de processos em deadlock, assim os outros processos conseguirão seus recursos. Para eliminar o processo, escolher um que possa ser reexecutado desde o início;

Evitar Dinamicamente: alocação cuidadosa de recursos, avaliando a chance de tal ação resultar ou não em deadlock;

Prevenção: é atacar uma das condições necessárias para se ter o estado de deadlock. Segue como acontece cada um dos possíveis ataques. Atacando:

- **Exclusão Mútua (usar spool em tudo):** recursos, como impressoras, podem fazer uso de spool: - o daemon de impressora é o único que usa o recurso impressora; - desta forma deadlock envolvendo a impressora é eliminado. Nem todos os dispositivos podem fazer parte de spool: Princípio: evitar alocar um recurso quando ele não for absolutamente necessário; tenta assegurar que o menor número possível de processos possa de fato requisitar o recurso.;
- **Posse e Espera (requisitar inicialmente todos os recursos necessários):** exige que todos os processos requisitem os recursos antes iniciarem, assim um processo nunca tem que esperar aquilo que precisa. O problema dessa mordomia é que podem não saber quantos e quais recursos vão precisar no início da execução e assim dimensionar recursos que outros processos poderiam estar usando. Uma variação desse tratamento é que o processo deveria desistir de todos os recursos para então requisitar todos os que são imediatamente necessário;
- **Não Preempção (retomar recursos alocados):** esta é uma opção inviável. considere um processo de posse da impressora, então no meio da impressão esse recurso é passado a outro processo, e num instante futuro retorna a impressora a força;
- **Espera Circular:** ordenação numérica dos recursos.

Existe também o cenário de Deadlock sem envolvimento de recursos. Isso é possível que dois processos entre em situação de deadlock se cada um estiver esperando

que o outro faça algo. Pode ocorrer com semáforos quando cada processo executa um down() sobre dois semáforos (mutéx, outro qualquer), se executados em ordem errada resulta em deadlock.

Para melhor entendimento, iremos ver alguns detalhes sobre **Processo**.

Fontes

<https://www.tecmundo.com.br/driver/207-o-que-e-driver-.htm>

<https://www.techtudo.com.br/dicas-e-tutoriais/noticia/2011/12/como-aumentar-memoria-virtual-no-windows-7.html>

<https://www.oficinadanet.com.br/post/12786-sistemas-operacionais-o-que-e-deadlock>

Professor Baully – aula 8