

Sistemas Operacionais - Exercícios

Assunto: Processos

1) Em um determinado sistema operacional X, os seguintes estados são definidos para os processos: executando (*running*), pronto (*ready*), bloqueado (*blocked*) e suspenso (*suspended*). Um processo está bloqueado se ele está esperando por uma permissão para usar o recurso, e ele está suspenso se ele está esperando por uma operação ser completada em um recurso que ele já adquiriu. Em sistemas operacionais Y esses dois estados são sobrepostos em um único estado bloqueado. Compare os méritos relativos às definições do sistema operacional X e dos SOs Y.

2) Supondo as seguintes definições:

blocked-suspended – é o estado onde são colocados os processos que estavam no estado *blocked* (pendente) quando foram retirados da memória para liberar espaço.

ready-suspended – é o estado para onde vão os processos retirados da memória quando estavam em qualquer um dos seguintes estados: *new*, *ready* ou *running*. Para este estado também são transferidos os processos que estão no estado *blocked-suspended* e tenha ocorrido o evento pelo qual estão pendentes.

Considere um SO cuja máquina de estados inclui os estados *ready* e *ready-suspended*. Suponha que seja hora do SO despachar um processo e que existam nesse momento processos tanto no estado *ready* como no estado *ready-suspended*, e que pelo menos um processo no estado *ready-suspended* possui prioridade maior do que qualquer processo no estado *ready*. Duas políticas extremas seriam: (a) sempre despachar um processo no estado *ready*, de forma a minimizar *swapping*; e (b) sempre dar preferência ao processo de mais alta prioridade, mesmo que isso possa significar a ocorrência de *swapping* quando este não é necessário.

Sugira uma política intermediária (descreva-a e crie um algoritmo em pseudo-código) que tente balancear prioridade e desempenho.