Capítulo 7 – Sincronização e Comunicação entre Processos

Preempção – sistema operacional interfere na execução de um processo.

(executando 🡪 PRONTO).

Escalonador FIFO (não preemptivo) – primeiro que entra é o primeiro a sair.

Escalonador CIRCULAR – cada processo tem uma determinada fatia de tempo (timeslice)

Preempção (por prioridade) – cada processo tem uma determinada prioridade.

TEMPO DE CPU – o tempo que um processo precisa para executar.

AGING – com o tempo, os processos vão ganhando idade e conseqüentemente também vão ganhando prioridade.

1 - É uma aplicação estruturada de maneira que partes diferentes do código do programa possam executar concorrentemente. Este tipo de aplicação tem como base a execução cooperativa de múltiplos processos ou threads, que trabalham em uma mesma tarefa na busca de um resultado comum.

3 - É impedir que dois ou mais processos acessem um mesmo recurso simultaneamente. Para isso, enquanto um processo estiver acessando determinado recurso, todos os demais processos que queiram acessá-lo deverão esperar pelo término da utilização do recurso.

6 - Essa solução apesar de simples, apresenta algumas limitações. Primeiramente, a multiprogramação pode ficar seriamente comprometida, já que a concorrência entre processos tem como base o uso de interrupções. Um caso mais grave poderia ocorrer caso um processo desabilitasse as interrupções e não tornasse a habilitá-las. Nesse caso, o sistema, provavelmente, teria seu funcionamento seriamente comprometido.

Em sistemas com múltiplos processadores, esta solução torna-se ineficiente devido ao tempo de propagação quando um processador sinaliza aos demais que as interrupções devem ser habilitadas ou desabilitadas. Outra consideração é que o mecanismo de clock do sistema é implementado através de interrupções, devendo esta solução ser utilizada com bastante critério.

7 - Na espera ocupada, toda vez que um processo não consegue entrar em sua região crítica, por já existir outro processo acessando o recurso, o processo permanece em looping, testando uma condição, até que lhe seja permitido o acesso. Dessa forma, o processo em looping consome tempo do processador desnecessariamente, podendo ocasionar problemas ao desempenho do sistema.

14 - a) Cliente A = 200 e Cliente B = 1.200

b) Cliente A = 400 e Cliente B = 1.100

Capítulo 8 – Gerência do Processador

1. Uma política de escalonamento é composta por critérios estabelecidos para determinar qual processo em estado de pronto será escolhido para fazer uso do processador.

2. O escalonador é uma rotina do sistema operacional que tem como principal função implementar os critérios da política de escalonamento. O dispatcher é responsável pela troca de contexto dos processos após o escalonador determinar qual processo deve fazer uso do processador.

3. Utilização do processador, throughput, tempo de Processador (tempo de UCP), tempo de espera, tempo de turnaround e tempo de resposta.

4. Tempo de processador ou tempo de UCP é o tempo que um processo leva no estado de execução durante seu processamento. Tempo de espera é o tempo total que um processo permanece na fila de pronto durante seu processamento, aguardando para ser executado. Tempo de turnaround é o tempo que um processo leva desde a sua criação até ao seu término, levando em consideração todo o tempo gasto na espera para alocação de memória, espera na fila de pronto (tempo de espera), processamento na UCP (tempo de processador) e na fila de espera, como nas operações de E/S. Tempo de resposta é o tempo decorrido entre uma requisição ao sistema ou à aplicção e o instante em que a resposta é exibida..

5. No escalonamento preemptivo, o sistema operacional pode interromper um processo em execução e passá-lo para o estado de pronto, com o objetivo de alocar outro processo na UCP. No escalonamento não-preemptivo, quando um processo está em execução, nenhum evento externo pode ocasionar a perda do uso do processador. O processo somente sai do estado de execução, caso termine seu processamento ou execute instruções do próprio código que ocasionem uma mudança para o estado de espera.

6. O FIFO é um escalonamento não-preemptivo onde o processo que chegar primeiro ao estado de pronto é o selecionado para execução. Este algoritmo é bastante simples, sendo necessária apenas uma fila, onde os processos que passam para o estado de pronto entram no seu final e são escalonados quando chegam ao seu início. Quando um processo vai para o estado de espera, o primeiro processo da fila de pronto é escalonado. Todos os processos quando saem do estado de espera entram no final da fila de pronto. O Circular é um escalonamento preemptivo, projetado especialmente para sistemas de tempo compartilhado. Esse algoritmo é bastante semelhante ao FIFO, porém, quando um processo passa para o estado de execução, existe um tempo limite para o uso contínuo do processador denominado fatia de tempo (time-slice) ou quantum.

7. No escalonamento SJF, o algoritmo de escalonamento seleciona o processo que tiver o menor tempo de processador ainda por executar. Dessa forma, o processo em estado de pronto que necessitar de menos tempo de UCP para terminar seu processamento é selecionado para execução. O escalonamento por prioridades é um escalonamento do tipo preemptivo realizado com base em um valor associado a cada processo denomidado prioridade de execução. O processo com maior prioridade no estado de pronto é sempre o escolhido para execução e processos com valores iguais são escalonados seguindo o critério de FIFO. Neste escalonamento, o conceito de fatia de tempo não existe, conseqüentemente, um processo em execução não pode sofrer preempção por tempo.

8. Preempção por tempo ocorre quando o sistema operacional interrompe o processo em execução em função da expiração da sua fatia de tempo, substituindo-o por outro processo. Preempção por prioridade, ocorre quando o sistema operacional interrompe o processo em execução em função de um processo entrar em estado de pronto com prioridade superior ao do processo em execução.

9. É um mecanismo onde o sistema operacional identifica o comportamento dos processos durante sua execução adaptando as políticas de escalonamento dinamicamente.

10. Escalonamento por prioridades onde é possível atribuir prioridades aos processos em função da sua importância. Além disso, o mecanismo de preempção por prioridades garante o escalonamento imediato de processos críticos quando esses passam para o estado de pronto.

11. Processos I/O-bound são favorecidos neste tipo de escalonamento. Como a probabilidade desse tipo de processo sofrer preempção por tempo é baixa, a tendência é que os processos I/O-bound permaneçam nas filas de alta prioridade enquanto os processos CPU-bound tendem a posicionar-se nas filas de prioridade mais baixa.