**Questionário de Sistemas Operacionais**

**Gustavo Soares Silva**

**1 - Quais as funções básicas da Gerência do Processador?**

Algumas delas são: manter o processador ocupado a maior parte do tempo, balancear o uso da UCP entre processos, privilegiar a execução de aplicações críticas, maximizar o throughput do sistema e oferecer tempos de resposta razoáveis para usuários interativos.

**2 - Liste os principais critérios de escalonamento.**

* Utilização do processador
* Throughput
* Tempo de processador/Tempo de UCP
* Tempo de espera
* Tempo de turnaround (*every now and then I get a little bit lonely || And you're never coming 'round - referência à Total eclipse of the hearth*)
* Tempo de resposta

**3 - O que são escalonamentos preemptivos e não-preemptivos?**

As políticas de escalonamento podem ser classificadas segundo a possibilidade de o sistema operacional interromper um processo em execução e substituí-lo por um outro, atividade esta conhecida como *preempção*. Sistemas operacionais que implementam escalonamento com *preempção* são mais complexos, ponim possibilitam políticas de escalonamento mais flexíveis.

O escalimarento nirpreemprivo lni o primeiro tipo de escalonamento implementada nos sistemas multiprogramáveis, onde predominava tipicamente o processa

mento butch. Nesse tipo de escalonamento, quando um processo está em execução, nenhum evento extremo pode ocasionar a perda do uso do processador. O processo imento sul do estado de exceção caso termine seu processamento ou execute ins truder do próprio código que ocasionou uma mudança para o estado de espera

O escalonamento preemptivo é caracterizado pela possibilidade do sistema operacional interromper um processo em execução e passá-lo para o estado de pronto, cum o objetivo de alocar outro processo na UCP. Com o uso da preempção, é possível ao Sistema priorizar a execução de processos como no caso de aplicações de tempo real onde o fator tempo é crítico. Outro benefício é a possibilidade de implementar políticas de escalonamento que compartilhem o processador de uma maneira mais uniforme, distribuindo de forma balanceada o uso da UCP entre os processos

Atualmente, a maioria dos sistemas operacionais possui políticas de escalonamento preemptivas que, apesar de tomarem os solos mais complexos, possibilitam a implementação dos diversos critérios de escalonamento apresentados.

**Caracterize os escalonamentos:**

**1 – FIFO:** No escalonamento first-in-first-out (FIFO scheduling), também conhecido como first-come-first-served (FCFS scheduling), o processo que chegar primeiro só estado de pronto é o selecionado para execução. Este algoritmo é bastante simples, sendo necessária apenas uma fila, onde os processos que passam para estado pronto entram no seu final e são escalonados quando chegam ao seu início. Quando o processo de execução termina seu processamento ou vai para o estado de espera, o primeiro processo da fila de pronto é escalonado. Quando saem do estado de espera, todos os processos entram no final da fila de pronto.

**2 – SJF:** No escalonamento shortest-job-first (SJF scheduling), também conhecido como shortest-process-neri (SPN scheduling), 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.

**3 – Cooperativo:** O escalonamento cooperativo é uma implementação que busca aumentar o grau de multiprogramação em políticas de escalonamentos que não possuam mecanismos de preempção, como o FIFO e o SJF não-preemptivo. Neste caso, um processo em execução pode voluntariamente liberar o processador retomando à fila de pronto, possibilitando que um novo processo seja escalonado e, assim, permitir melhor distribuição no uso do processador

A principal característica do escalonamento cooperativo está no fato de a liberação do processador ser uma tarefa realizada exclusivamente pelo processo em execução que de uma maneira cooperativa libera a UCP para um outro processo. Neste mecanismo, o processo em execução verifica periodicamente uma fila de mensagens para determinar se existem outros processos na fila de pronto