**SISTEMAS OPERACIONAIS**

1. Quais são as diferenças essenciais entre um aplicativo comum (por exemplo, o Microsoft Word) e um sistema operacional?

*- Programas aplicativos: programas que utilizam os recursos computacionais para resolver problemas dos usuários.*

*- Sistema operacional: controla e coordena o uso do hardware pelos diversos programas aplicativos de diversos usuários.*

1. Por que um sistema operacional é necessário?

*Para permitir o compartilhamento dos recursos de hardware por diversas aplicações e para facilitar o desenvolvimento de novas aplicações, deixando o gerenciamento dos recursos de hardware a cargo do sistema operacional.*

1. Que nome é dado às chamadas realizadas pelas aplicações aos serviços disponibilizados pelo sistema operacional? Mencione como exemplo duas destas chamadas.

*Chamadas de sistemas (System calls):*

*- connect: inicia uma conexão em um soquete*

*- fork: cria um novo processo*

1. O que é um processo?

*Processo é um "trabalho" em execução em um sistema computacional.*

1. Um processador com um único núcleo executa vários processos simultaneamente? Explique sua resposta.

*Normalmente há muito mais processos do que processadores, portanto, os processos precisam compartilhar os processadores. Para que sejam executados, cada processo recebe uma parcela do tempo de execução no processador.*

1. O que ocorre quando o sistema operacional faz o chaveamento entre processos?

*O sistema operacional salva todo o estado do processo que está em execução e carrega o estado do processo que entrará em execução.*

1. Qual é a diferença entre thread e processo?

*- Thread: execução de uma sequência de instruções de um processo.*

*- Processo: uma tarefa em execução no sistema que pode conter várias threads.*

1. Qual é a diferença entre escalonamento preemptivo e não-preemptivo.

*- Escalonamento preemptivo: o sistema operacional pode "tomar" o processador, mesmo que o processo ainda tenha instruções a executar. Ocorre como eventos externos ao processo que está em execução.*

*- Escalonamento não-preemptivo: o processador não pode ser "tomado" do processo. Dependem somente do próprio processo.*

1. Cite 2 algoritmos de escalonamento de processos e explique o funcionamento de cada um.

*- Escalonamento First Come, First Served: os processos são executados na ordem em que chegam ao escalonador. Pode apresentar diferentes tempos médios de espera de acordo com a ordem de chegada dos processos.*

*- Escalonamento Shortest Job First: minimiza o tempo de espera para um conjunto de processos. Associa a cada processo o seu próximo tempo de CPU, e prioriza o processo na fila de pronto com o menor tempo.*

1. Cite 2 recursos disponibilizados pelo sistema operacional para ajudar o programador a implementar programas concorrentes sem interferência entre threads, e diga em qual situação é mais conveniente utilizar cada recurso.

*- Semáforo: mecanismo que garante que somente N processos podem acessar um certo recurso em um dado momento. É conveniente usar quando um processo depende do outro*

*- Passagem de mensagens: combinação de transferência de dados com exclusão mútua e sincronização.*