Sistemas Operacionais Lista de Exercícios

Conceitos de Sistemas Operacionais

Charles Ferreira

1. **Defina com suas palavras o que é um sistema operacional e qual a sua importância dentro da computação.**

Resp.: Um programa que age como um intermediário entre usuário e o hardware. A sua importância consiste em, por exemplo, executar programas do usuário, tornar a solução de problemas mais fácil, tornar os sistemas computacionais convenientes ao uso, usar o hardware do computador de maneira eficiente.

1. **O que é multiprogramação? O que é concorrência? Diferencie as duas situações.**

Resp.: Multiprogramação é a execução de múltiplos programas. Um único programa não pode ocupar CPU e dispositivos todo o tempo, então enquanto um programa espera outro pode ser executado.

Concorrência é quando dispositivos de entrada e saída (E/S) e CPU podem executar concorrentemente.

1. **Qual a diferença entre o modo kernel e o modo usuário? É necessário existir tais modos?**

Resp.: A diferença é que algumas instruções são projetadas como privilegiadas, sendo somente executáveis em modo kernel, sendo assim o usuário tem acesso apenas as instruções “comuns”.

Sim, é necessário existir ambos os modos já que iso possibilita determinar quando o sistema está executando código do usuário ou do kernel além de possibilitar que o sistema operacional se proteja e proteja outros componentes do sistemas que não devem ser acessados por usuários comuns (modo usuário).

1. **Das instruções a seguir, quais só podem ser executadas em modo kernel?**

(a)Desabilitar todas as instruções

* 1. Ler o horário do relógio

(c)Alterar o mapa de memória

(d)Executar programas

1. **Qual a finalidade de uma chamada de sistema (system call) em um sistema operacional?**

Resp.: System calls são a interface entre um processo e o sistema operacional, dessa forma é feita a passagem de parâmetro entre o processo do usuário e o s.o;

Através dele (dos comandos) é que podemos alternar entre modo kernel e modo usuário, fazemos o gerenciamento de processos, memória, arquivos, diretórios, gerenciamento de tempo, etc;

1. **Qual o propósito de uma interrupção? Qual a diferença entre uma trap e uma interrupção? As traps podem ser geradas intencionalmente pelo programa do usuário?**

Resp.: Para controlar entrada e saída de dados, não é interessante que a CPU tenha que ficar continuamente monitorando e status de dispositivos como discos ou teclados. O mecanismo de interrupções permite que o hardware "chame a atenção" da CPU quando há algo a ser feito. Logo a interrupção é uma mudança de fluxo (interrupção gerada por hardware) já a trap é uma interrupção gerada por software;

1. **Explique e diferencie o que é um programa, um processo e uma thread.**
2. **Qual a vantagem e desvantagem de cada um destes cenários**:

(a)programa multi-processos

* 1. programa multi-thread

1. **Descreva dois exemplos de programas nos quais a programação multithread não oferece melhor performace do que um programa single-thread.**
2. **Por que é importante que o escalonador de processos saiba diferenciar processos que**

**sejam I/O Bound e CPU bound?**

Resp.:

1. **Qual o significado de espera ocupada? Que outros tipos de espera existem no sistema operacional? A espera ocupada pode ser evitada? Qual as dificuldades envolvidas?**

Resp.:  espera ocupada ou espera ociosa é uma técnica em que um processo verifica uma condição repetidamente até que ela seja verdadeira, como esperar o pressionamento da tecla de um teclado ou a espera da liberação de recurso. Também pode ser usada para gerar atrasos na execução do sistema por um período determinado de tempo. No geral é considerada um anti-padrão e deve ser evitada, já que o tempo de CPU gasto poderia ser usado noutra tarefa.

Um exemplo de uso seria no escritor/leitor, onde é necessária a espera ocupada (por exemplo wait() e notify() do Java) para permitir que esse processo aconteça.

Uma dificuldade é a sincronização de processos, já que caso aconteça erro durante a sincronização a thread pode travar e o programa nunca terá fim. Um dos problemas seria, novamente, escritor/leitor e jantar dos filósofos;

1. **Explique a diferença entre fragmentação interna e fragmentação externa**

Resp.: A**fragmentação interna ocorre na alocação de memória de tamanho fixo, enquanto a fragmentação externa ocorre na alocação de memória dinâmica**. Um exemplo de alocação dinâmica seria o maloc da linguagem C;

1. **Quais foram as estratégias de alocação de memória discutidas em aula?**

Resp.: First Fit (desvantagem: fragmentação interna), Best Fit e Worst Fit

1. **Compare as estratégias de organização da memória principal utilizadas pela alocação contígua, segmentação e paginação considerando os seguintes aspectos:**

(a)fragmentação externa

(b)fragmentação interna

1. **Muitos sistemas permitem que um programa aloque mais memória no seu espaço de endereçamento durante a execução. Alocação de dados na heap dos programas é um exemplo de tal alocação. O que é necessário para permitir alocação de memória dinâmica nos seguintes esquemas:**

(a)alocação contígua de memória

(b)segmentação

(c)paginação