



Curso: Engenharia de Sistemas

Aula 6: Sistemas Operacionais Atividade de Pesquisa

Data: 11/09/2023

Valor: 1,0

Tipo de Aula : Prática

Estratégia: Atividade em duplas- EVA

Crítérios de Avaliação: Observação direta, Resolução de problemas, raciocínio

Data Entrega: 11/09/2023 - prazo final : 18/09/2023

Informações:

- Trabalho em Duplas – identificar o trabalho com nome dos alunos
 - A atividade para ser feita em aula
 - Postar no EVA, ou arquivo no pendrive. Poderá ser enviado por e-mail: flavio_matheus@yahoo.com.br
-

Conceitos de sistemas operacionais

A maioria dos sistemas operacionais fornece determinados conceitos e abstrações básicos, como processos, espaços de endereços e arquivos, que são fundamentais para compreendê-los.

Processos

Um conceito fundamental em todos os sistemas operacionais é o **processo**. Um processo é basicamente um programa em execução. Associado a cada processo está o **espaço de endereçamento**, uma lista de posições de

memória que vai de 0 a algum máximo, onde o processo pode ler e escrever. O espaço de endereçamento contém o programa executável, os dados do programa e sua pilha. Também associado com cada processo há um conjunto de recursos, em geral abrangendo registradores (incluindo o contador de programa e o ponteiro de pilha), uma lista de arquivos abertos, alarmes pendentes, listas de processos relacionados e todas as demais

informações necessárias para executar um programa.

Um processo é na essência um contêiner que armazena todas as informações necessárias para executar um programa.

Sistemas operacionais mais sofisticados permitem que múltiplos programas estejam na memória ao mesmo tempo. Para evitar que interfiram entre si (e com o sistema operacional), algum tipo de mecanismo de proteção é necessário.

Espaços de endereçamento

Uma questão diferente relacionada à memória, mas igualmente importante, é o gerenciamento de espaços de endereçamento dos processos. Em geral, cada processo tem algum conjunto de endereços que ele pode usar, tipicamente indo de 0 até algum máximo. No caso mais simples, a quantidade máxima de espaço de endereços que um processo tem é menor do que a memória principal.

Dessa maneira, um processo pode preencher todo o seu espaço de endereçamento e haverá espaço suficiente na memória principal para armazená-lo inteiramente. No entanto, em muitos computadores os endereços são de 32 ou 64 bits, dando um espaço de endereçamento de 2^{32} e 2^{64} , respectivamente. O que acontece se um processo tem mais espaço de endereçamento do que o computador tem de memória principal e o processo quer usá-lo inteiramente?

Nos primeiros computadores, ele não teria sorte. Hoje, existe uma técnica chamada memória virtual, como já mencionado, na qual o sistema operacional mantém parte do espaço de endereçamento na memória principal e parte no disco, enviando trechos entre eles para lá e para cá conforme a necessidade. Na



Curso: Engenharia de Sistemas

Aula 6: Sistemas Operacionais Atividade de Pesquisa

Data: 11/09/2023

Valor: 1,0

Tipo de Aula : Prática

Estratégia: Atividade em duplas- EVA

Crítérios de Avaliação: Observação direta, Resolução de problemas, raciocínio

essência, o sistema operacional cria a abstração de um espaço de endereçamento como o conjunto de endereços ao qual um processo pode se referir.

Arquivos

Outro conceito fundamental que conta com o suporte de virtualmente todos os sistemas operacionais é o sistema de arquivos. Como já foi observado, uma função importante do sistema operacional é esconder as peculiaridades dos discos e outros dispositivos de E/S e apresentar ao programador um modelo agradável e claro de arquivos que sejam independentes dos dispositivos.

Chamadas de sistema são obviamente necessárias para criar, remover, ler e escrever arquivos. Antes que um arquivo possa ser lido, ele deve ser localizado no disco e aberto, e após ter sido lido, deve ser fechado, assim as chamadas de sistema são fornecidas para fazer essas coisas. Para fornecer um lugar para manter os arquivos, a maioria dos sistemas operacionais de PCs tem o conceito de um **diretório** como uma maneira de agrupar os arquivos.

Entrada/Saída

Todos os computadores têm dispositivos físicos para obter entradas e produzir saídas. Afinal, para que serviria um computador se os usuários não pudessem dizer a ele o que fazer e não pudessem receber os resultados após ele ter feito o trabalho pedido? Existem muitos tipos de dispositivos de entrada e de saída, incluindo teclados, monitores, impressoras e assim por diante. Cabe ao sistema operacional gerenciá-los. Em consequência, todo sistema operacional tem um subsistema de E/S para gerenciar os dispositivos de E/S.

Alguns softwares de E/S são independentes do dispositivo, isto é, aplicam-se igualmente bem a muitos ou a todos dispositivos de E/S.

O interpretador de comandos (shell)

O sistema operacional é o código que executa as chamadas de sistema. Editores, compiladores, montadores, ligadores (*linkers*), programas utilitários e interpretadores de comandos definitivamente não fazem parte do sistema operacional, mesmo que sejam importantes e úteis.

Correndo o risco de confundir as coisas de certa maneira, nesta seção examinaremos brevemente o interpretador de comandos UNIX, o shell. Embora não faça parte do sistema operacional, ele faz um uso intensivo de muitos aspectos do sistema operacional e serve assim como um bom exemplo de como as chamadas de sistema são usadas. Ele também é a principal interface entre um usuário sentado no seu terminal e o sistema operacional, a não ser que o usuário esteja usando uma interface de usuário gráfica. Muitos shells existem, incluindo, *sh*, *csh*, *ksh* e *bash*. Todos eles dão suporte à funcionalidade descrita a seguir, derivada do shell (*sh*) original.

Memória virtual

A memória virtual (discutida no Capítulo 3) proporciona a capacidade de executar programas maiores do que a memória física da máquina, rapidamente movendo pedaços entre a memória RAM e o disco. Ela passou por um desenvolvimento similar, primeiro aparecendo nos computadores de grande porte, então passando para os minis e os micros. A memória virtual também permitiu que um programa se conectasse dinamicamente a uma biblioteca no momento da execução em vez de fazê-lo na compilação. O MULTICS foi o primeiro sistema a permitir isso. Por fim, a ideia propagou-se adiante e agora é amplamente usada na maioria dos sistemas UNIX e Windows.



Curso: Engenharia de Sistemas

Aula 6: Sistemas Operacionais Atividade de Pesquisa

Data: 11/09/2023

Valor: 1,0

Tipo de Aula : Prática

Estratégia: Atividade em duplas- EVA

Crítérios de Avaliação: Observação direta, Resolução de problemas, raciocínio

A API Win32 do Windows

Um programa UNIX consiste de um código que faz uma coisa ou outra, fazendo chamadas de sistema para ter determinados serviços realizados. Em comparação, um programa Windows é normalmente direcionado por eventos. O programa principal espera por algum evento acontecer, então chama uma rotina para lidar com ele. Eventos típicos são teclas sendo pressionadas, o mouse sendo movido, um botão do mouse acionado, ou um disco flexível inserido.

Tratadores são então chamados para processar o evento, atualizar a tela e o estado do programa interno. Como um todo, isso leva a um estilo de certa maneira diferente de programação do que com o UNIX, mas tendo em vista que o foco deste livro está na função e estrutura do sistema operacional, esses modelos de programação diferentes não nos dizem mais respeito.

A Microsoft definiu um conjunto de rotinas chamadas de API Win32 (Application Programming Interface — interface de programação de aplicativos) que se espera que os programadores usem para acessar os serviços do sistema operacional. Essa interface tem contado com o suporte (parcial) de todas as versões do Windows desde o Windows 95. Ao desacoplar a interface API das chamadas de sistema reais, a Microsoft retém a capacidade de mudar as chamadas de sistema reais a qualquer tempo (mesmo de um lançamento para outro) sem invalidar os programas existentes. O que de fato constitui o Win32 também é um tanto ambíguo, pois versões recentes do Windows têm muitas chamadas novas que não estavam disponíveis anteriormente. Nesta seção, Win32 significa a interface que conta com o suporte de todas as versões do Windows. A Win32 proporciona compatibilidade entre as versões do Windows.

Trabalho - valor 1,0

1. Quais são as duas principais funções de um sistema operacional?
2. A ideia de família de computadores foi introduzida na década de 1960 com os computadores de grande porte System/360 da IBM. Essa ideia está ultrapassada ou ainda é válida? – comente
3. Qual é a diferença entre sistemas de compartilhamento de tempo e de multiprogramação?
4. Há várias metas de projeto na construção de um sistema operacional, por exemplo, utilização de recursos, oportunidade, robustez e assim por diante. Dê um exemplo de duas metas de projeto que podem contradizer uma à outra.
5. Qual é a diferença entre modo núcleo e modo usuário? Explique como ter dois modos distintos ajuda no projeto de um sistema operacional.
6. Qual tipo de multiplexação (tempo, espaço ou ambos) pode ser usado para compartilhar os seguintes recursos: CPU, memória, disco, placa de rede, impressora, teclado e monitor?