

**Educação
Profissional
Paulista**

Técnico em
**Ciência de
Dados**

Introdução à Ciência de Dados, ferramentas e jargões da área

Introdução à informática

Aula 1

[DADOS]ANO1C1B2S11A1

Exposição



Objetivo da Aula

Introduzir os conceitos de sistemas operacionais.



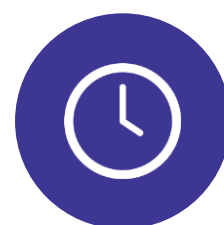
Competências da Unidade (Técnicas e Socioemocionais)

- Compreender e dominar os principais conceitos e noções de sistemas operacionais na informática;
- Analisar situações em que cada tipo de sistema operacional pode ser utilizado e identificar os principais equipamentos e finalidades de aplicação para seus usos.



Recursos Didáticos

- Recurso audiovisual para a exibição de textos, vídeos e imagens;
- Artigos selecionados para leitura.



Duração da Aula

50 minutos.

Exposição



© Getty Images

Sistemas operacionais – Revisão

Definição de um sistema operacional

Um sistema operacional (SO) é um software que atua como intermediário entre o usuário e o hardware do computador.

Ele gerencia os recursos do sistema, proporcionando um ambiente no qual os programas de aplicativo podem ser executados.

Pontos-chave:

- Interface entre o usuário e o hardware;
- Gerencia e aloca recursos como CPU, memória e dispositivos de armazenamento;
- Facilita a criação e a execução de programas de software.

Paginação

Paginação é uma técnica essencial no gerenciamento de memória de sistemas operacionais modernos, otimizando o uso de maneira eficiente.

Nesse método, a memória é **dividida em blocos de tamanho fixo**, como em páginas, que são armazenadas na RAM e no disco rígido, na área de *swap*.

Entre os benefícios, estão:

- Criar um espaço de endereçamento virtual contínuo para processos;
- Possibilidade de execução de múltiplos processos simultâneos sem conflito;
- Maximização do uso da memória.

Paginação

A gestão das páginas entre a memória e o disco é realizada a partir das tabelas de páginas, que gerenciam a correspondência entre páginas virtuais e endereços físicos.

A ***Translation Lookaside Buffer (TLB)***, um cache avançado, agiliza esse processo ao armazenar as traduções mais recentes ou acessadas, reduzindo o tempo para converter endereços virtuais em físicos.



Tome nota

A TLB é vital para o sistema. Sem ela, o acesso constante às tabelas de páginas pode sobrecarregar o sistema.

Exposição

Exemplos de paginação

Exemplo 1:

Em um sistema com **4 GB de RAM**, um processo pode ter um espaço de endereço virtual muito maior, como 32GB, graças à paginação.

As partes do programa que estão sendo usadas são mantidas na RAM, enquanto o resto permanece no disco até ser necessário.

Exemplo 2:

Sistemas como **Windows 11 e Linux** usam um método chamado paginação para administrar a memória do computador de modo eficaz. Imagine que cada programa é um livro.



© Getty Images

Exemplos de paginação

Exemplo 3:

A **TLB**, em CPUs modernas, como os processadores Intel Core i7, acelera o mapeamento de endereços virtuais para físicos.

Então, ocorre o melhor desempenho da paginação ao armazenar as traduções recentes.



Tome nota

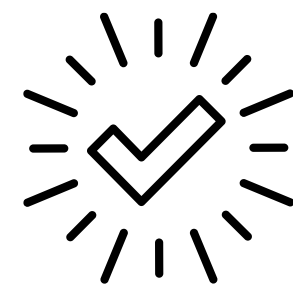
A **paginação** permite que o computador use apenas as páginas necessárias do livro em vez de abri-lo inteiro. Isso economiza espaço e possibilita que mais programas rodem ao mesmo tempo.

Swapping e thrashing

Swapping: processo de transferir páginas ou segmentos de memória entre RAM e o disco para otimizar o uso de memória pelos processos.

Utilização: técnica usada para liberar memória para processos que necessitam de mais recursos ou para balancear o uso da memória entre os processos em execução.

O uso excessivo do *swapping* pode causar **thrashing**, um estado no qual o sistema gasta mais tempo com essa transferência do que executando os processos.



Swapping e thrashing são conceitos relacionados ao gerenciamento de memória em sistemas operacionais.

Ambos são usados quando a memória está limitada ou estão sendo intensamente utilizados.

Swapping e thrashing

O **thrashing** pode afetar negativamente o desempenho do sistema, causando lentidão e possíveis travamentos.

Como minimizar o thrashing?

Sistemas operacionais **aprimoram o algoritmo**, que decide quais páginas trocar, e controlam a quantidade de programas rodando ao mesmo tempo por meio da multiprogramação.

Isso ajuda a evitar sobrecarga por processos simultâneos.

Além disso, preveem quais páginas serão necessárias, mantêm as mais relevantes na memória e utilizam algoritmos avançados de substituição, como o Least Recently Used (LRU).



Tome nota

Least Recently Used (LRU) é um algoritmo de substituição de cache usado para gerenciar a memória de forma eficiente, decidindo qual item em memória será substituído para dar espaço a um novo item.

Exposição



© Getty Images

Exemplos de *swapping* e *thrashing*

Exemplo 1:

Um sistema operacional pode usar algoritmos como **LRU** (***Least Recently Used***) para prevenir o *thrashing*, substituindo as páginas que não foram usadas por um longo período, minimizando o impacto no desempenho.

Exemplo 2:

Um servidor web sob alta carga pode começar a fazer ***swapping intensivo*** quando a RAM fica cheia, movendo dados ativos entre a memória e o disco.

Isso pode resultar em uma desaceleração significativa do tempo de resposta às requisições.

Exemplos de *swapping* e *thrashing*

Exemplo 3:

Aumentar a quantidade de RAM em um computador pode reduzir a frequência de *swapping* e **prevenir o *thrashing***.

Isso é observado em máquinas de edição de vídeo, nas quais grandes quantidades de dados precisam ser processadas rapidamente.



Exemplos

Exemplos populares de software de servidor web incluem Apache HTTP Server, Nginx, Microsoft Internet Information Services (IIS) e LiteSpeed.

Alocação de memória

A alocação de memória é vital no gerenciamento de sistemas operacionais, pois define a distribuição da memória para os processos e busca o uso eficiente, reduzindo a fragmentação. Estratégias como **best-fit**, **worst-fit** e **first-fit** têm prós e contras, conforme a situação.

<i>best-fit</i>	<i>worst-fit</i>	<i>first-fit</i>
Aloca o menor bloco capaz de atender à demanda, minimizando a fragmentação externa, mas pode ser mais lenta, devido à necessidade de procurar o bloco ideal.	Opta pelo maior bloco disponível, mantendo grandes blocos de memória livres para uso futuro, mas pode levar a uma maior fragmentação com o tempo.	Aloca o primeiro bloco de memória grande, oferecendo um bom equilíbrio entre performance e simplicidade.

Elaborado especialmente para o curso.

Alocação de memória

Sistemas operacionais utilizam técnicas para lidar com a **fragmentação**, a fim de garantir que os recursos de memória sejam otimizados entre os processos, como:

- compactação da memória;
- uso de algoritmos de alocação que equilibram eficiência e velocidade.

Tipos de fragmentação:

Fragmentação interna: ocorre quando o espaço de memória alocado é maior do que o necessário pelo processo, levando a um desperdício de memória nos blocos designados.

Fragmentação externa: ocorre quando há espaço livre na memória total, mas está fracionado em blocos pequenos demais para atender a uma solicitação de alocação.

Exposição

Exemplos de alocação de memória

Exemplo 1:

No Linux, o algoritmo de **alocação first-fit** é bastante usado para alocar espaço de memória para novos processos, procurando o primeiro espaço que seja grande o suficiente para acomodar a solicitação.

Exemplo 2:

A **fragmentação interna** pode ocorrer quando, em sistemas com páginas fixas de memória, o espaço não é totalmente usado pelos processos, como se uma página tivesse 4 KB, mas o processo só usasse 3,5 KB, sobrando 0,5 KB inutilizado.



© Getty Images

Exemplos de alocação de memória

Exemplo 3:

Em sistemas como o Windows, que usam alocação dinâmica de memória, pode surgir **fragmentação externa**. Isso ocorre quando "buracos" pequenos demais para uso efetivo aparecem após aplicativos serem desinstalados ou alocações irregulares de memória serem feitas.



Tome nota

Essa situação pode requerer uma **desfragmentação**, para otimizar o espaço.

Vamos
fazer um
quiz

O que a TLB armazena?

Endereços físicos
somente.

Endereços virtuais
somente.

Traduções de
endereços recentes.

Dados de usuário e
kernel.



Vamos
fazer um
quiz

O que a TLB armazena?



**Endereços físicos
somente.**

**Endereços virtuais
somente.**



**Traduções de
endereços recentes.**

**Dados de usuário e
kernel.**



FEEDBACK GERAL DA ATIVIDADE

A *Translation Lookaside Buffer* (TLB) armazena as traduções mais recentes ou frequentemente acessadas de endereços virtuais para físicos, o que acelera significativamente o processo de tradução de endereços durante a paginação.



Vamos
fazer um
quiz

O que o *thrashing* causa?

Melhoria do
desempenho.

Aumento de espaço
em disco.

Desaceleração
do sistema.

Redução de uso
de CPU.



Vamos
fazer um
quiz

O que o *thrashing* causa?



**Melhoria do
desempenho.**

**Aumento de espaço
em disco.**



**Desaceleração
do sistema.**

**Redução de uso
de CPU.**



FEEDBACK GERAL DA ATIVIDADE

O *thrashing* ocorre quando há excesso de *swapping* (troca de páginas de memória com o disco), levando a uma desaceleração significativa no sistema, pois o sistema operacional passa mais tempo paginando do que executando processos.



Vamos
fazer um
quiz

Qual alocação minimiza fragmentação externa?

Best-fit.

Worst-fit.

First-fit.

Random-fit.



Vamos
fazer um
quiz

Qual alocação minimiza fragmentação externa?



Best-fit.

Worst-fit.



First-fit.

Random-fit.



FEEDBACK GERAL DA ATIVIDADE

O algoritmo de alocação *best-fit* procura o menor bloco de memória livre que é suficiente para atender à demanda, minimizando assim a quantidade de espaço desperdiçado e a fragmentação externa.



O que nós
aprendemos
hoje?

© Getty Images

Hoje desenvolvemos:

- 1** A habilidade no uso da **paginação** para gerir memória eficientemente, entendendo a divisão do espaço em páginas e o papel crucial da **TLB** na aceleração do acesso à memória.
- 2** Os conhecimentos de **swapping e thrashing**, compreendendo o intercâmbio de páginas de memória com o disco e as estratégias para evitar a degradação do desempenho por excesso de paginação.
- 3** A expertise nas estratégias de alocação de memória, como **best-fit e first-fit**, e de como minimizar a fragmentação, optando pela melhor conformidade com o espaço disponível.



Saiba mais

Neste artigo, vamos fazer o download e a instalação do Ubuntu 18.04, vamos falar de algumas distribuições do Linux e listar alguns dos principais comandos.

DELVA, P. Linux: download, instalação, distribuições e principais comandos. *Alura*, 2020. Disponível em: <https://www.alura.com.br/artigos/linux-download-instalacao-distribuicoes-e-principais-comandos>. Acesso em: 11 mar. 2024.

Referências da aula

DELVA, P. Linux: download, instalação, distribuições e principais comandos. *Alura*, 2020. Disponível em: <https://www.alura.com.br/artigos/linux-download-instalacao-distribuicoes-e-principais-comandos>.

Acesso em: 11 mar. 2024.

TANENBAUM, A. S.; BOS, H. *Sistemas operacionais modernos*. São Paulo: Pearson Education do Brasil, 2016.

Identidade visual: Imagens © Getty Images

**Educação
Profissional
Paulista**

Técnico em
**Ciência de
Dados**