

# Implementação de Núcleo de Sistema Operacional

Revisão SO - Parte 1

Prof. Dr. Denis M. L. Martins

Engenharia de Computação: 7° Semestre

### Objetivos de Aprendizado



- Explicar o conceito de Sistema Operacional e seus serviços típicos.
- Explicar o conceito de kernel, incluindo a API de chamadas de sistema, modo usuário e modo kernel.

#### Disclaimer



Parte do material apresentado a seguir foi adaptado de *IT Systems – Open Educational Resource*, disponível em https://oer.gitlab.io/oer-courses/it-systems/, produzido por Jens Lechtenböger, e distribuído sob a licença CC BY-SA 4.0.

## Sistema Operacional: Definição e Fundamentos



#### Software que:

- utiliza recursos de hardware de um sistema computacional, e
- provê suporte para execução de outros softwares.

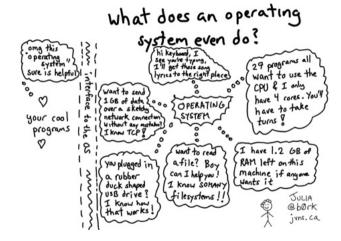


Figura 1: O que um SO faz. Créditos: Julia Evans.



- Gerenciamento de multitarefa: O sistema operacional permite a execução simultânea de múltiplas computações, gerenciando a alternância entre elas e garantindo a retomada correta de cada uma.
- Controle de concorrência: Regula a interação entre processos concorrentes, impedindo acessos indevidos a estruturas de dados e fornecendo áreas de memória isoladas para diferentes computações.
- Interação entre computações assíncronas: Suporta a troca de informações entre computações que não são executadas ao mesmo tempo, por meio de sistemas de arquivos e armazenamento de longo prazo.
- Interação via rede: Facilita a comunicação entre computações distribuídas em diferentes sistemas computacionais através de redes, sendo um recurso essencial em sistemas operacionais modernos.



- Gerenciamento de multitarefa: O sistema operacional permite a execução simultânea de múltiplas computações, gerenciando a alternância entre elas e garantindo a retomada correta de cada uma.
- Controle de concorrência: Regula a interação entre processos concorrentes, impedindo acessos indevidos a estruturas de dados e fornecendo áreas de memória isoladas para diferentes computações.
- Interação entre computações assíncronas: Suporta a troca de informações entre computações que não são executadas ao mesmo tempo, por meio de sistemas de arquivos e armazenamento de longo prazo.
- Interação via rede: Facilita a comunicação entre computações distribuídas em diferentes sistemas computacionais através de redes, sendo um recurso essencial em sistemas operacionais modernos.



- Gerenciamento de multitarefa: O sistema operacional permite a execução simultânea de múltiplas computações, gerenciando a alternância entre elas e garantindo a retomada correta de cada uma.
- Controle de concorrência: Regula a interação entre processos concorrentes, impedindo acessos indevidos a estruturas de dados e fornecendo áreas de memória isoladas para diferentes computações.
- Interação entre computações assíncronas: Suporta a troca de informações entre computações que não são executadas ao mesmo tempo, por meio de sistemas de arquivos e armazenamento de longo prazo.
- Interação via rede: Facilita a comunicação entre computações distribuídas em diferentes sistemas computacionais através de redes, sendo um recurso essencial em sistemas operacionais modernos.



- Gerenciamento de multitarefa: O sistema operacional permite a execução simultânea de múltiplas computações, gerenciando a alternância entre elas e garantindo a retomada correta de cada uma.
- Controle de concorrência: Regula a interação entre processos concorrentes, impedindo acessos indevidos a estruturas de dados e fornecendo áreas de memória isoladas para diferentes computações.
- Interação entre computações assíncronas: Suporta a troca de informações entre computações que não são executadas ao mesmo tempo, por meio de sistemas de arquivos e armazenamento de longo prazo.
- Interação via rede: Facilita a comunicação entre computações distribuídas em diferentes sistemas computacionais através de redes, sendo um recurso essencial em sistemas operacionais modernos.



- O núcleo (kernel) de um SO oferece uma API (Application Programming Interface)
  - Expõe um conjunto de interfaces para os serviços do OS (system calls).
  - Deixa transparente para o programador uma série de detalhes (operações) de baixo nível.
- Interface com o usuário (UI) é um item não obrigatório.
  - ▶ UI: processos usando funcionalidade do núcleo do SO para gerenciar entrada do usuário, iniciar programas, produzir saída, ...
  - Exemplos de UIs: linha de comando, Explorer (Windows), ambientes de desktop para GNU/Linux, assistentes virtuais.

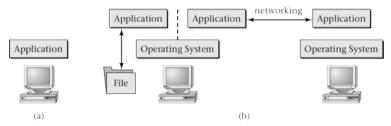


Figura 2: Em (a), baixo nível de abstração. Em (b), alto nível de abstração. Imagem: Max Hailperin.

#### Como falar com o SO



- Chamada de sistema = função = parte da API do kernel
- Implementação de serviços do sistema operacional, como:
  - Execução de processos
  - Alocação de memória principal
  - Acesso a recursos de hardware (exemplo: teclado, rede, arquivos e disco, placa de vídeo)
- Diferentes sistemas operacionais oferecem diferentes chamadas de sistema (ou seja, APIs incompatíveis)
  - Com diferentes implementações
  - Com diferentes convenções de chamada



https://wizardzines.com/comics/how-to-talk-to-your-operating-system.

Figura 3: System calls. Imagem: Julia Evans.

## Espaço de Núcleo versus Espaço de Usuário



O.15 user 0.73 system

time spent by

the kernel doing

work for your

Drocess

time spent in

your process

- No espaço de núcleo (kernel space), o SO tem controle total sobre o hardware.
- Aplicações rodando em espaço do usuário precisam invocar chamadas de sistema: requisitar ao SO para realizar alguma tarefa que requer maiores privilégios (e.g., receber input de algum hardware/aparelho ou escrever um arquivo).
- System calls levam a mudanças de contexto entre diferentes contextos de execução. (Vamos explorar esse conceito mais à frente no curso).

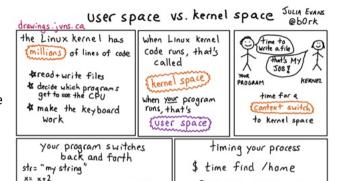


Figura 4: User space vs. kernel space. Imagem: Julia Evans.

user space

file write (str) - x switch to

str= str + y

#### Núcleo de SO e suas variantes



- SO roda como qualquer outro programa na CPU.
- O núcleo contém a parte mais central de um SO.
  - Código + dados do núcelo reside normalmente em memória principal.
  - As funcionalidades do núcleo rodam na CPU em kernel mode, reagindo a system calls e interrupções (tema de aula futura)
- Variantes:
  - Monolítico: núcleo único com todos os serviços integrados
  - ▶ **Microkernel:** núcleo mínimo, com serviços em espaço de usuário
  - ► **Híbrido:** mistura microkernel e monolítico para otimizar desempenho.

#### Núcleo de SO e suas variantes



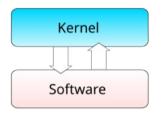


Figura 5: Núcleo monolítico. Imagem: Wikipedia

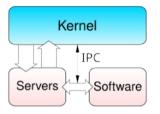


Figura 6: Micronúcleo. Imagem: Wikipedia

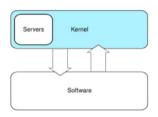


Figura 7: Núcleo híbrido. Imagem: Wikipedia

### Linha do tempo



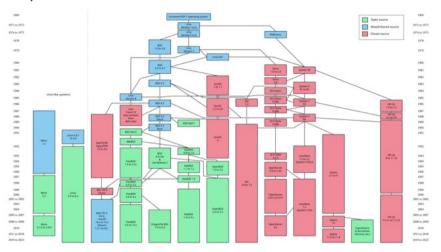


Figura 8: Linha to tempo de sistemas Unix-like. Imagem: Wikipedia

#### Tamanho de um SO



- O código fonte do GNU/Linux tem cerca de 5 milhões de linhas.
- Já o código fonte do Windows, com seus pacotes essenciais teria cerca de 70 milhões de linhas.
- Como esse código pode ser mantido e compreendido?



Figura 9: Mapa do núcleo do Linux. Imagem: Costa Shulyupin.

#### Tamanho de um SO



- O código fonte do GNU/Linux tem cerca de 5 milhões de linhas.
- Já o código fonte do Windows, com seus pacotes essenciais teria cerca de 70 milhões de linhas.
- Como esse código pode ser mantido e compreendido?
  Através de abstração, modularização e organização em camadas.



Figura 9: Mapa do núcleo do Linux. Imagem: Costa Shulyupin.

#### Conteúdo relacionado



- YouTube #1: What is a Kernel? do canal Techquickie.
- YouTube #2: How Does Linux Boot Process Work? do canal ByteByteGo.
- Ler as seções 1.1 (O que é um sistema operacional?) e 1.2 (História dos sistemas operacionais) do livro TANENBAUM, A.; Sistemas Operacionais Modernos. 4a ed. Pearson Brasil, 2015.



#### Questão 1

Quais afirmações são corretas sobre conceitos de Sistemas Operacionais:

- a) O sistema operacional gerencia a execução de aplicações em termos de **threads**.
- b) O sistema operacional cria uma nova thread para cada chamada de sistema (system call).
- c) O sistema operacional agenda (schedule) threads para execução nos núcleos da CPU.
- d) O sistema operacional cria novas threads para utilizar todos os núcleos da CPU.
- e) O time-slicing cria a ilusão de paralelismo em núcleos de CPU únicos.
- f) O sistema operacional bloqueia recursos compartilhados para evitar anomalias de atualização.



#### Questão 1

Quais afirmações são corretas sobre conceitos de Sistemas Operacionais:

- a) O sistema operacional gerencia a execução de aplicações em termos de **threads**.
- b) O sistema operacional cria uma nova thread para cada chamada de sistema (system call).
- c) O sistema operacional agenda (schedule) threads para execução nos núcleos da CPU.
- d) O sistema operacional cria novas threads para utilizar todos os núcleos da CPU.
- e) O time-slicing cria a ilusão de paralelismo em núcleos de CPU únicos.
- f) O sistema operacional bloqueia recursos compartilhados para evitar anomalias de atualização.

Resposta: São verdadeiras: (a), (c), (e).



#### Questão 2

Quais afirmações são corretas sobre conceitos de Sistemas Operacionais:

- a) O código do kernel é armazenado na **ROM**.
- b) As aplicações podem acessar mais memória RAM se precisarem.
- c) O sistema operacional gerencia as aplicações como processos.
- d) O sistema operacional aloca memória RAM como parte da **memória virtual**.
- e) As threads de um processo compartilham recursos.
- f) O sistema operacional protege arquivos com criptografia.



#### Questão 2

Quais afirmações são corretas sobre conceitos de Sistemas Operacionais:

- a) O código do kernel é armazenado na ROM.
- b) As aplicações podem acessar mais memória RAM se precisarem.
- c) O sistema operacional gerencia as aplicações como processos.
- d) O sistema operacional aloca memória RAM como parte da memória virtual.
- e) As threads de um processo compartilham recursos.
- f) O sistema operacional protege arquivos com criptografia.

Resposta: São verdadeiras: (c), (d), (e).



## Dúvidas e Discussão

Prof. Dr. Denis M. L. Martins denis.mayr@puc-campinas.edu.br