



Universidade Federal de Lavras  
PPGCC  
PCC508 – Sistemas Operacionais

## **Lista Avaliativa 1**

Douglas Aquino T. Mendes  
18 de outubro de 2024

# Sumário

<b>1</b>	<b>Perguntas</b>	<b>2</b>
1.1	O que é um sistema operacional? . . . . .	2
1.2	O que o modo supervisor e usuário? . . . . .	2
1.3	Quais os passos para a realizar uma chamada de sistema? . . . . .	2
1.4	O que é o sistema de proteção de arquivos? . . . . .	3
1.5	O que é um processo? . . . . .	3
1.6	O que é espaço de endereçamento e memória virtual? . . . . .	3
1.7	Como estruturar um sistema operacional de micronúcleo? . . . . .	4
<b>2</b>	<b>Desenvolver um programa</b>	<b>4</b>
2.1	Sorteia um número entre 0 e 200 . . . . .	4
2.2	Imprimirá números sequenciais em intervalos de 1 segundo . . . . .	4

# 1 Perguntas

## 1.1 O que é um sistema operacional?

**Pergunta:** Explique o que é um sistema operacional, utilizando a visão do sistema como uma máquina estendida e como um gerenciador de recursos.

**Resposta:** Sistema operacional **como máquina estendida** é uma abstração para facilitar a comunicação do software com o hardware, tendo em vista que, gerenciar as informações diretamente no hardware é algo muito complexo, então após definir e implementar as abstrações de um SO (Sistema Operacional) o sistema operacional esconde as complexidades do hardware, oferecendo interfaces mais simples e eficientes para os programadores resolverem os problemas. Já como um **gerenciador de recursos**, o sistema operacional é responsável por alocar e monitorar o uso dos recursos do sistema, como CPU, memória e dispositivos de E/S (entrada e saída). Ele decide quais programas podem acessar determinados recursos, controla o uso desses recursos, contabiliza seu consumo e resolve conflitos de requisição quando há várias solicitações simultâneas. [1]

## 1.2 O que o modo supervisor e usuário?

**Pergunta:** Explique o que o modo supervisor e usuário, e em que momentos existe a troca de modo do processador.

**Resposta:** O modo supervisor e o modo usuário são dois estados de execução do processador que permitem ao sistema operacional controlar o acesso aos recursos do sistema e garantir a segurança e estabilidade da execução dos programas. No modo supervisor (também conhecido como modo kernel ou privilegiado), o processador tem acesso irrestrito a todos os recursos do hardware, como a memória, dispositivos de E/S e instruções privilegiadas. Esse modo é usado pelo sistema operacional para realizar operações críticas, como gerenciamento de memória, controle de processos e manipulação direta de hardware. No modo usuário, o processador tem acesso restrito, ou seja, os programas de aplicação podem executar apenas um subconjunto limitado de instruções, sem acesso direto a recursos de hardware sensíveis. Isso previne que falhas ou erros em aplicações comprometam a estabilidade e segurança do sistema [2] [3].

A troca de modo do processador ocorre em momentos, como:

- *System calls* (chamadas de sistema): Quando um programa em modo usuário precisa de algum serviço do sistema operacional, ele solicita uma chamada de sistema, o que aciona a troca para o modo supervisor para que o SO execute a operação.
- Interrupções: Quando um dispositivo de hardware, como um disco ou teclado, precisa da atenção do processador, uma interrupção ocorre, levando o processador a trocar para o modo supervisor para que o sistema operacional trate essa interrupção.
- Troca de contexto: Durante a troca entre processos ou threads, o sistema operacional precisa executar operações em modo supervisor para gerenciar a memória e o estado do processador antes de retornar ao modo usuário.

## 1.3 Quais os passos para a realizar uma chamada de sistema?

**Pergunta:** Explique, em detalhes, quais são os passos para a realização de uma chamada de sistema.

**Resposta:**

1. Para obter serviços do sistema operacional, um programa de usuário deve fazer uma chamada de sistema, que, por meio de uma instrução TRAP, chaveia do modo usuário para o modo núcleo e passa o controle para o sistema operacional.

2. A execução do programa muda de modo usuário para modo kernel. Isso é feito para garantir que o código do sistema operacional tenha controle sobre os recursos do sistema e possa executar operações que não são permitidas em modo usuário.
3. O número da chamada de sistema, que é geralmente passado em um registrador, é usado pelo kernel para identificar qual serviço deve ser executado.
4. O kernel verifica a validade dos parâmetros fornecidos.
5. O kernel executa a operação correspondente à chamada de sistema (como ler um arquivo, criar um processo ou alocar memória).
6. Após a execução da chamada de sistema, o kernel prepara o resultado da operação, que pode incluir um valor de retorno, como um identificador de arquivo, um ponteiro de memória ou um código de erro.
7. O sistema operacional muda novamente o modo de execução de volta para modo usuário, permitindo que o aplicativo continue sua execução.
8. O aplicativo lê o resultado da chamada de sistema, que geralmente está disponível em um registrador ou em um local de memória específico.
9. O programa retoma sua execução a partir do ponto onde a chamada de sistema foi feita, agora com os resultados da operação. [3]

## 1.4 O que é o sistema de proteção de arquivos?

**Pergunta:** Explique o sistema de proteção de arquivos baseado em um código de 9 bits para cada arquivo.

**Resposta:** Quando um arquivo é criado, ele recebe o UID e GID do processo criador, além de um conjunto de permissões que determinam o acesso do proprietário, do grupo do proprietário e dos demais usuários. As permissões são organizadas em três categorias: leitura (r), escrita (w) e execução (x). Para arquivos executáveis, a permissão de execução é relevante, e tentar executar um arquivo não executável resultará em erro. Como existem três categorias de usuários e três tipos de permissão por categoria, utiliza-se 9 bits para representar os direitos de acesso [3]. Por exemplo, se o dono do arquivo tem permissões de leitura, escrita e execução, o grupo tem permissão de leitura e execução, e outros têm apenas permissão de leitura, o código de 9 bits seria “rwxr-xr-”.

## 1.5 O que é um processo?

**Pergunta:** O que é um processo? Quando um processo não está em execução, onde as informações sobre o mesmo são armazenadas para permitirem que ele continue posteriormente sua execução do mesmo ponto onde foi parada?

**Resposta:** Um processo é um programa em execução, contendo um espaço de endereçamento que inclui o código executável, dados e pilha. Ele também possui recursos como registradores, lista de arquivos abertos, alarmes e informações necessárias para sua execução. Quando um processo é suspenso, todas essas informações são salvas, permitindo que ele seja retomado do ponto onde parou. Essas informações são armazenadas na tabela de processos, que contém o estado dos registradores e outros dados essenciais para reiniciar o processo posteriormente. O espaço de endereçamento, conhecido como imagem do núcleo, também faz parte desse contexto [3].

## 1.6 O que é espaço de endereçamento e memória virtual?

**Pergunta:** Explique o que é espaço de endereçamento e memória virtual.

**Resposta:** **Espaço de Endereçamento** refere-se ao conjunto de endereços de memória que um processo pode acessar. É a faixa de endereços de memória que vai de um valor mínimo (geralmente 0) até um valor

máximo, definida pelo sistema operacional e pela arquitetura do hardware. Esse espaço contém tudo o que o processo necessita para executar: o código do programa (instruções), dados (variáveis) e a pilha (para gerenciar chamadas de função e variáveis locais). Cada processo tem seu próprio espaço de endereçamento, isolado dos demais, garantindo segurança e estabilidade no sistema. E a **Memória Virtual** é uma técnica de gerenciamento de memória usada pelos sistemas operacionais que proporciona a capacidade de executar programas maiores do que a memória física da máquina, rapidamente movendo pedaços entre a memória RAM e o disco. Na memória virtual, o espaço de endereçamento de um processo é dividido em blocos chamados de páginas, e essas páginas podem ser armazenadas tanto na RAM quanto no disco. Quando uma página que um processo precisa não está na RAM, ocorre uma interrupção chamada page fault, e o sistema operacional carrega a página correspondente do disco para a memória, possibilitando a execução contínua do processo [3].

## 1.7 Como estruturar um sistema operacional de micronúcleo?

**Pergunta:** Explique uma forma de estruturar um sistema operacional chamada de micronúcleo.

**Resposta:** Para estruturar um sistema operacional baseado em micronúcleo, a ideia central é dividir o sistema em dois componentes principais, o Micronúcleo que é o núcleo mínimo que executa em modo privilegiado (kernel mode) e gerencia apenas as funcionalidades essenciais do sistema, e os serviços de usuário (que normalmente estariam no núcleo em um sistema monolítico), que operam em modo não privilegiado (user mode).

O micronúcleo deve ser responsável por executar apenas as operações essenciais que exigem acesso direto ao hardware, como, gerenciamento de processos, gerenciamento de memória, comunicação entre processos e gerenciamento de interrupções. As funcionalidades adicionais (como por exemplo, gerenciamento de arquivos, gerenciamento de dispositivos, serviços de rede e gerenciamento de interface) que em sistemas monolíticos estariam no kernel, são implementadas como processos separados no espaço de usuário. Cada serviço possui responsabilidade limitada e comunicação com outros serviços via IPC.

Como o micronúcleo executa apenas o mínimo necessário, ele deve fornecer mecanismos eficientes de IPC para que os processos possam trocar informações de forma segura e rápida. E como os serviços são processos de usuário isolados, falhas em um serviço não devem derrubar o sistema como um todo mas o micronúcleo precisa fornecer mecanismos para detectar falhas em serviços de usuário e reiniciar automaticamente um serviço de usuário que falhou sem comprometer o restante do sistema. [3] [4] [5]

## 2 Desenvolver um programa

### 2.1 Sorteia um número entre 0 e 200

**Enunciado:** Desenvolver um programa onde um processo pai cria cinco filhos. Cada um dos filhos sorteia um número entre 0 e 200 e envia para o pai através de um pipe. O processo pai imprime então qual foi o menor número recebido e também o PID do filho correspondente ao menor número enviado.

**Resposta:** O código fonte está no arquivo `atv8.c` e pode ser compilado com o comando `"gcc atv8.c -o atv8"` e executado com o comando `"./atv8"`

### 2.2 Imprimirá números sequenciais em intervalos de 1 segundo

**Enunciado:** Um processo pai irá criar 3 filhos. Cada filho imprimirá na tela números sequenciais em intervalos de 1 segundo, juntamente com o seu PID. O processo pai, após a criação dos filhos, irá permitir que somente 1 filho execute em cada momento por um período de 10s. A cada período, o filho que está executando será parado com `SIGSTOP` e o próximo será liberado para executar (com `SIGCONT`). Não esquecer que após a criação dos 3 filhos, todos devem ser parados para dar após início a essa sequência.

**Resposta:** O código fonte está no arquivo `atv8.c` e pode ser compilado com o comando `"gcc atv9.c -o atv9"` e executado com o comando `"./atv9"`

## Referências

- [1] Departamento de Eletrônica e Computação (UFSM), *Sistemas Operacionais*. Santa Maria: Universidade Federal de Santa Maria (UFSM), 2018, Disponível em: [https://www.ufsm.br/app/uploads/sites/413/2018/12/08\\_sistemas\\_operacionais.pdf](https://www.ufsm.br/app/uploads/sites/413/2018/12/08_sistemas_operacionais.pdf). Acesso em: 16 out. 2024.
- [2] Grupo de Teleinformática e Automação (UFRJ). “Definições e Conceitos sobre Sistemas Operacionais.” Acesso em: 16 out. 2024. (2008), endereço: [https://www.gta.ufrj.br/grad/08\\_1/virtual/DefinieSeConceitos.html](https://www.gta.ufrj.br/grad/08_1/virtual/DefinieSeConceitos.html).
- [3] A. S. Tanenbaum e H. Bos, *SISTEMAS OPERACIONAIS MODERNOS*, 4<sup>a</sup> ed. Boston: Pearson, 2021.
- [4] C. Maziero, *Sistemas Operacionais: Conceitos e Mecanismos*, Acessado em: 17 out. 2024, 2016. endereço: <https://wiki.inf.ufpr.br/maziero/lib/exe/fetch.php?media=socm:socm-livro.pdf>.
- [5] A. Leite, *Aula 03 - Sistemas Operacionais: Estrutura de Sistemas Operacionais (Micronúcleos)*, Acessado em: 17 out. 2024, 2020. endereço: <http://www.univasf.edu.br/~andreza.leite/aulas/S0/Aula03.pdf>.