

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
BAHIA
Campus Salvador

Aula 1: Introdução aos Sistemas Operacionais

Instituto Federal da Bahia

INF009 - Sistemas Operacionais

Prof^a Flávia Maristela

Pensando melhor...

- Porque cursar a disciplina de S.O.?
 - A disciplina é obrigatória!
 - A disciplina é ótima!
 - Sistemas operacionais é fácil! (**Definitivamente, NÃO!**)
 - A disciplina vai ser importante para mim!
- Na maioria das vezes, COMPUTAÇÃO é muito mais **transpiração** do que inspiração...
:-)

O que veremos nesta aula?

- Introdução
- Histórico
- Principais funções de um sistema operacional

- Estrutura de sistemas computacionais
- Sistema operacional na visão do usuário
- Visão interna de um sistema operacional

Mas afinal, para que serve um sistema operacional?



A história dos Sistemas Operacionais

- Se confunde com as Gerações de Computadores:
 - 1ª. Geração de SO: programadores desenvolviam o programa e as funções de gerência para cada uma das aplicações fazia parte do código da própria aplicação;
 - 2ª. Geração: o sistema de processamento em lote (armazenamento prévio de diversas tarefas para processamento sequencial no computador). Representou um marco para a geração;

A história dos Sistemas Operacionais

- Se confunde com as Gerações de Computadores:
 - 3^a. Geração: desenvolvimento do conceito de multiprogramação (particionamento da memória do computador em diversas parcelas para que múltiplas tarefas pudessem ser executadas), dando a sensação para o usuário de um paralelismo não existente;
 - 4^a. Geração: marcada pelo desenvolvimento de interfaces gráficas. As redes de computadores impulsionam o surgimento de sistemas operacionais de rede e os sistemas operacionais distribuídos.

Infográfico

- <http://www.tecmundo.com.br/sistema-operacional/2031-a-historia-dos-sistemas-operacionais-ilustracao-.htm>

Visão geral de um sistema computacional

- Um sistema computacional consiste em:
 - Programas Aplicativos
 - Programas do Sistema
 - Dispositivos de Hardware



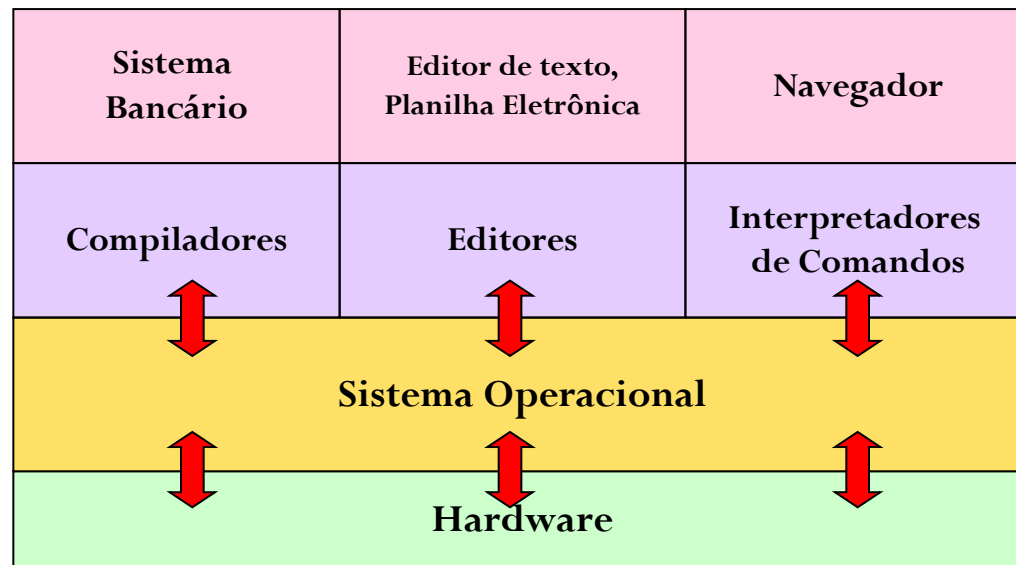
Sistema Bancário	Editor de texto, Planilha Eletrônica	Navegador
Compiladores	Editores	Interpretadores de Comandos

É necessário um componente capaz de...

- Gerenciar diferentes recursos
 - CPU, memória, disco, ...
- Melhorar o desempenho do computador
 - tempo de resposta, *throughput**,...
- Prover uma arquitetura que facilite a programação:
 - É possível abstrair a camada de hardware!

É necessário um componente capaz de...

- Servir de interface entre o *hardware* e o resto do computador



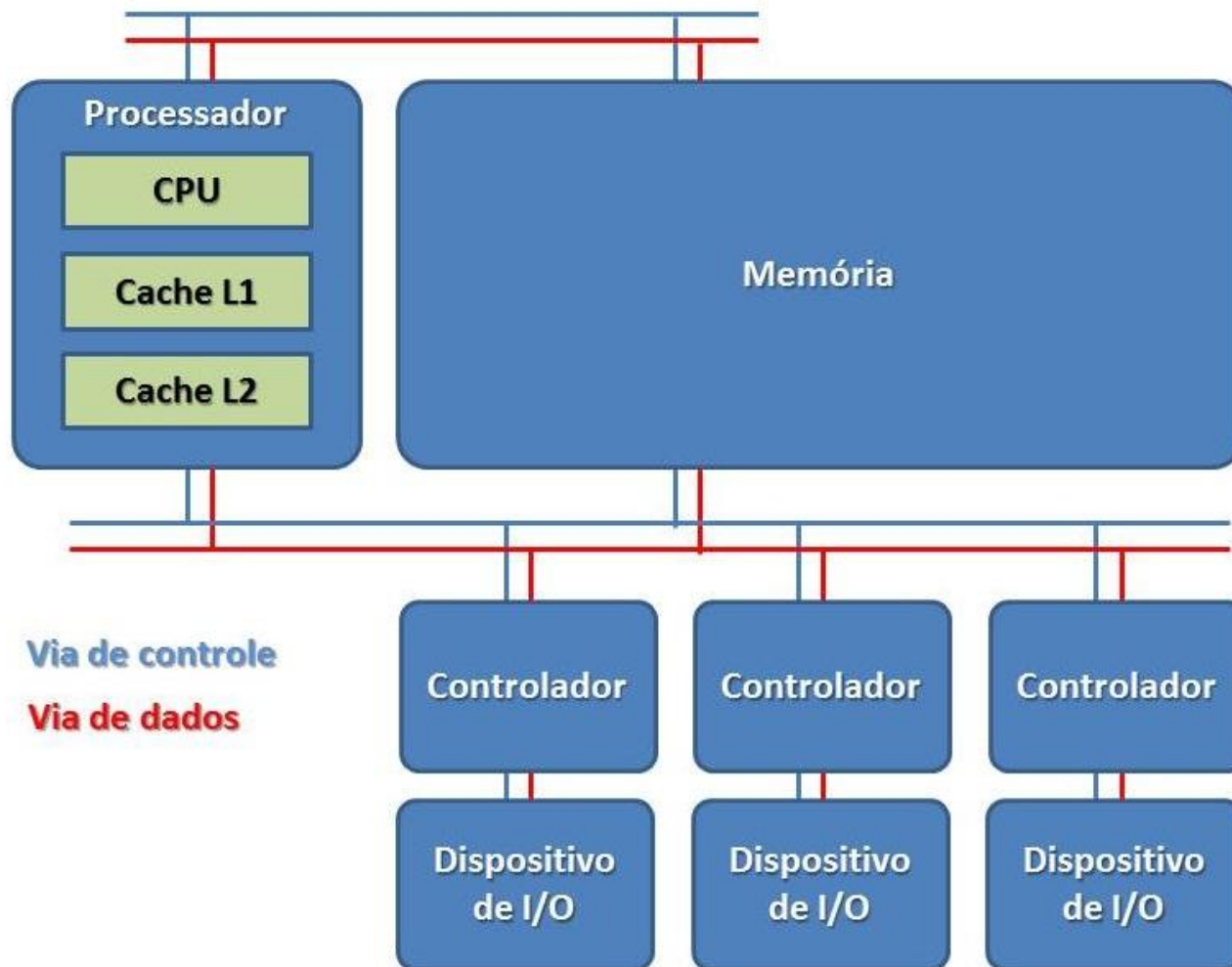
Quais as principais funções de um sistema operacional?

- Gerenciar recursos de hardware
 - Gerenciamento de Processador (Gerência de Processos)
 - Gerenciamento de Memória
 - Gerenciamento de E/S
- Gerenciar recursos de Software:
 - Programas
 - Arquivos
- A gerência de recursos implica em garantir que os mesmos sejam utilizados sem que ocorram erros, que possam levar a falhas no sistema como um todo.

Gerenciando o hardware :: o processador

- O que faz o processador?
 - Responsável pelas operações lógicas e aritméticas do computador
- O processador busca suas instruções para execução na memória
- Instruções podem ser executadas de duas formas:
 - Modo *kernel* (privilegiado)
 - Modo usuário

Arquitetura de Von Neumann



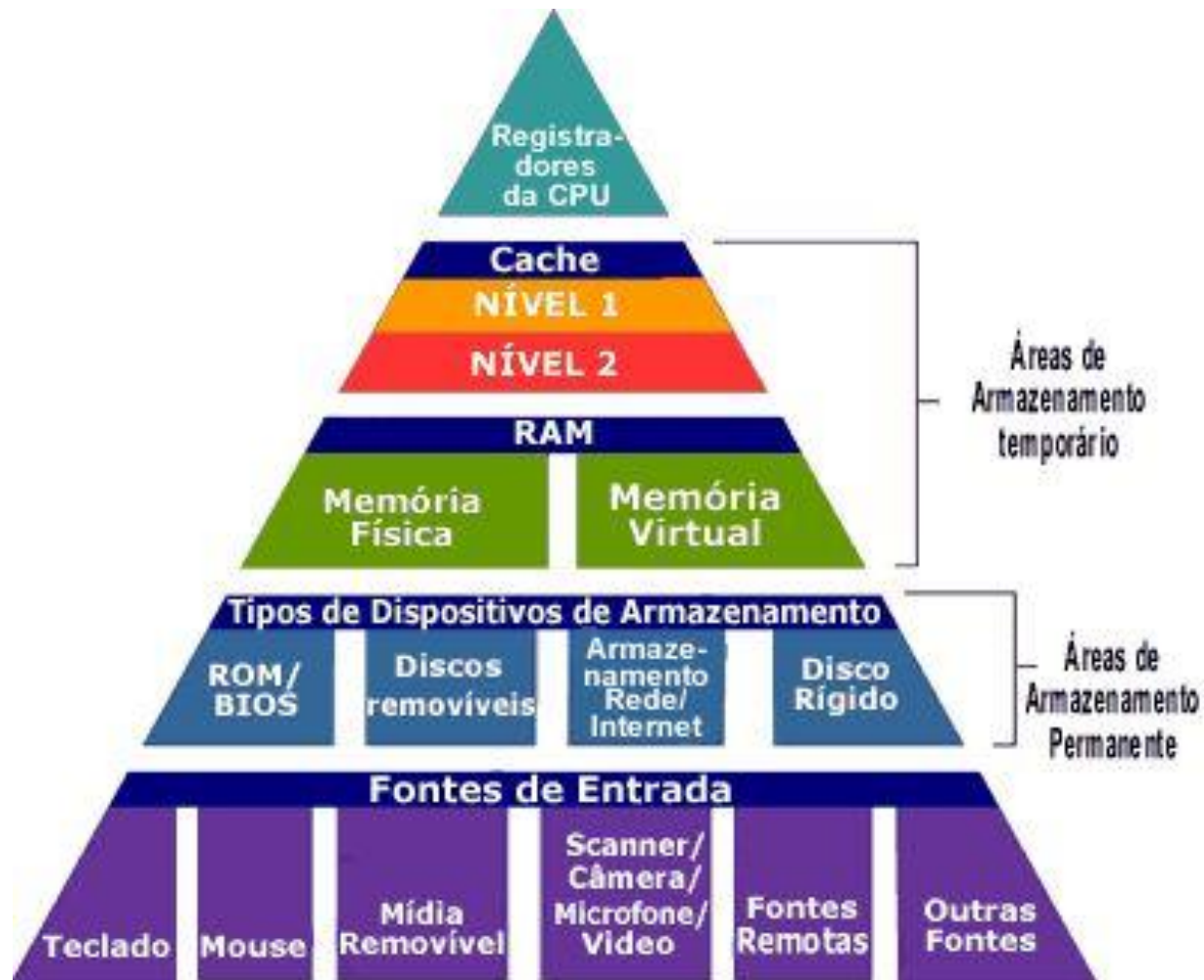
Gerenciando o hardware :: o processador

- *Modo kernel*
 - CPU pode executar qualquer instrução
 - CPU tem acesso direto ao hardware
- *Modo usuário*
 - Permite a execução de apenas um subconjunto de instruções
 - Acesso limitado aos atributos das instruções
 - Funções que envolvem E/S e proteção de memória não são acessíveis

Gerenciando o hardware :: a memória

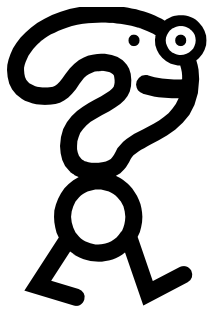
- Para que serve a memória?
- Segundo principal componente do computador.
- Tipos de Memória:
 - Cache
 - Principal
 - Disco Magnético
 - Fita Magnética

Hierarquia de Memória



Gerenciando o hardware :: Entrada e Saída

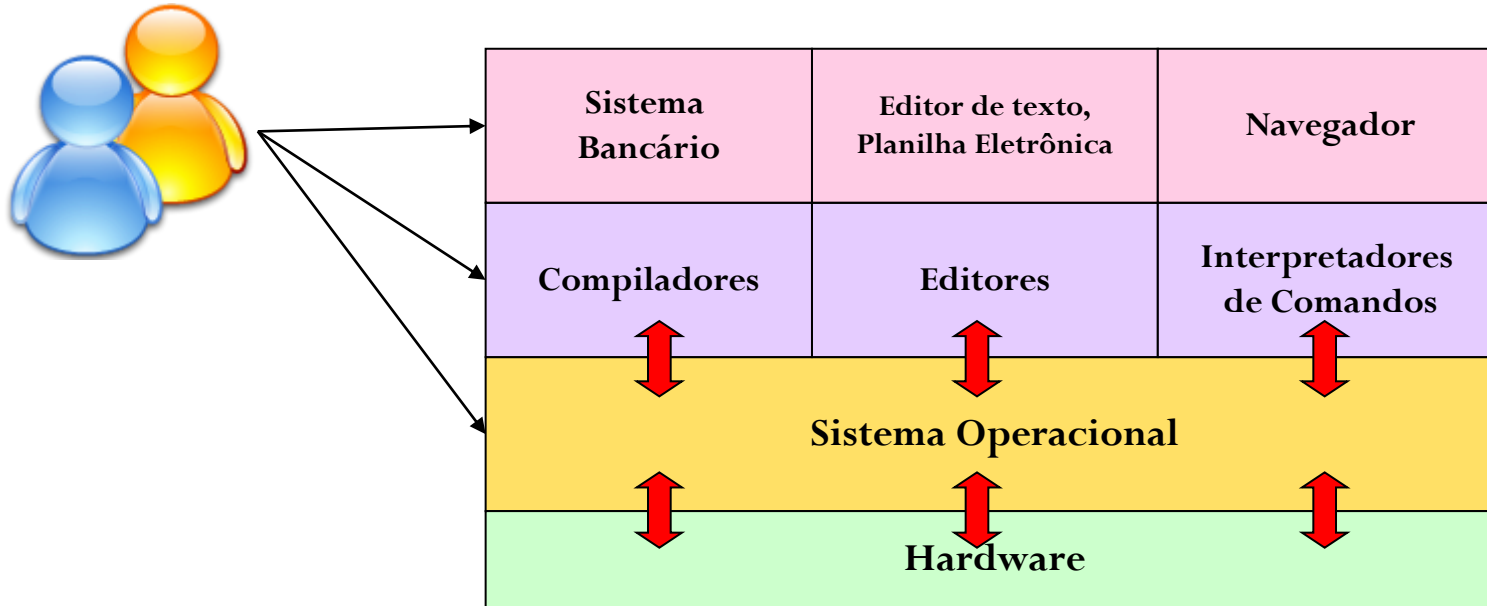
- Dispositivos de E/S interagem fortemente com o sistema operacional
- Modos de realizar E/S:
 - Espera ociosa
 - Interrupção
 - *Direct Memory Access* (DMA)
- Vocês se lembram o que é isto?
 - **Vale a pena relembrar!**



Para pensar um pouco...

- Existe algum programa que precisa executar no modo *kernel*?
- Que programas executam no modo usuário?
- Como um programa do modo usuário pode acessar os dispositivos de E/S?

Como o usuário acessa o hardware?



- Programas de sistema ou utilitários e aplicativos
 - Controle permanece com o usuário
- Chamadas de sistema (*system calls*)
 - Controle é passado para o sistema operacional

De fora para dentro...

- Sistema operacional na visão do usuário:
 - system calls:
 - usadas quando programas solicitam serviços ao sistema operacional;
 - transferem a execução do programa para o sistema operacional;
 - São executadas em modo *kernel*
 - Exemplo: um usuário deseja ler o conteúdo de um arquivo texto.

#Algoritmo

- 1. usuário acessa o editor*
- 2. editor faz uma chamada de sistema para o sistema operacional*
- 3. sistema operacional verifica se o arquivo existe*
- 4. se arquivo existe*
- 5. chamada de sistema para ler conteúdo do arquivo*
- 6. chamada de sistema para exibir conteúdo no terminal*
- 7. senão*
- 8. Sistema operacional informa ao editor que arquivo não existe*

Outros tipos de sistemas operacionais

- Bart Simpson resolveu comprar um computador e criar um sistema operacional apenas para usar o Bloco de Notas. **Que características esse Sistema Operacional precisaria ter?**

Meu sistema operacional vai ser monousuário e monotarefa!



Outros tipos de sistemas operacionais

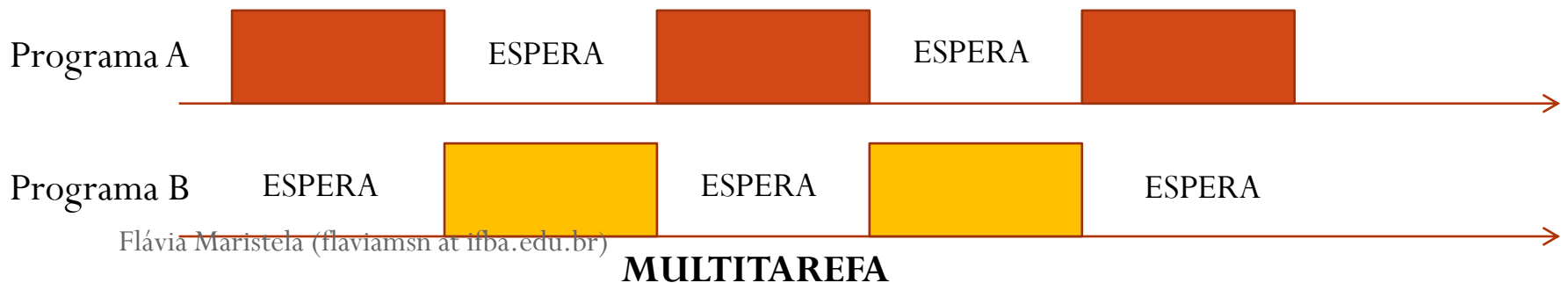
- Mais tarde, Bart resolveu que gostaria também de ouvir música, enquanto editava seu texto. **Que mudanças ele precisaria fazer no Sistema Operacional?**

Meu sistema operacional vai precisar ser alterado para considerar várias tarefas sendo executadas!



Outros tipos de sistema operacional

- Sistema Monotarefa vs. Sistema Multitarefa
 - Monotarefa: admite a execução de apenas uma tarefa por vez (monousuário e monoprogramado)
 - Multitarefa: mais de uma tarefa é executada por vez (monousuário ou multiusuário, multiprogramado)



Outros tipos de sistema operacional

- Com relação ao usuário
 - Monousuário vs. Multiusuário
- Com relação ao número de tarefas que podem ser executadas:
 - Monotarefa vs. Multitarefa
- Com relação ao tipo de programação
 - Monoprogramação vs. Multiprogramação



Qual a relação entre estes conceitos?

Outros tipos de Sistema

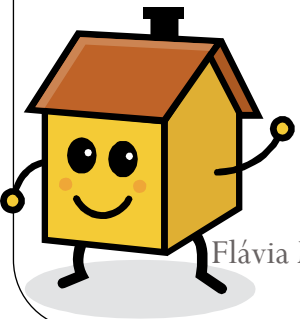
- **Sistemas Monoprocessados**
 - Reconhecem e gerenciam apenas uma unidade de processamento (CPU)
- **Sistemas Multiprocessados:**
 - São capazes de gerenciar mais de uma unidade de processamento (CPU)

O que vimos nesta aula?

- Conceito de sistemas operacionais
- Principais funções dos sistemas operacionais
- Visão Geral das gerências:
 - Processador
 - Memória
 - E/S (revisão)
- Classificação dos sistemas operacionais:
 - Quanto ao número de usuários
 - Quanto a quantidade de processos executados
 - Quanto ao tipo de programação
 - Quanto ao número de unidades de processamento

Outros tipos de sistema operacional

- Sistemas Multiprogramáveis / Multitarefa
 - Batch
 - Time-sharing
 - Distribuídos
 - Tempo Real
 - Embarcados



Para os alunos:

Pesquisem quais as principais características destes tipos de sistema operacional para a **próxima aula**.

Mais algumas perguntas!

- O computador realiza acessos simultâneos ao processador, memória ou dispositivos de E/S?
- Como é possível controlar o acesso simultâneo ao processador?
 - Escalonamento
- Como é possível controlar o acesso simultâneo a memória?
 - Segmentação
- Como é possível controlar o acesso simultâneo aos dispositivos de E/S?
 - Interrupção

Dúvidas?



Um presente para os alunos!

1. O que é um sistema operacional?
2. Porque os sistemas operacionais foram criados?
3. Quais as principais funções de um sistema operacional?
4. O que é kernel?
5. Quais os tipos de acesso a hardware? Quais as diferenças entre eles?
6. O que são as interrupções? Para que servem?
7. O que são as *system calls*?
8. Quais os tipos de sistema operacional? Explique cada um deles.
9. Caracterize as abordagens: *kernel* monolítico vs. *kernel* em camadas.