

DCA-108 Sistemas Operacionais

Luiz Affonso Guedes
www.dca.ufrn.br/~affonso
affonso@dca.ufrn.br



Apresentação

- Disciplina básica do curso de Engenharia de Computação
- Carga-horária: 60h teóricas
- Associadas com a disciplina Programação Concorrente
 - 30h predominantemente de laboratório
- Pré-requisito:
 - Organização/arquitetura de computadores
- Horário e Local:
 - 24M56
 - Sala 4A8
- Avaliação:
 - 3 provas teóricas
 - 2 trabalhos em equipe

Objetivo da Disciplina

- Apresentar os conceitos básicos de sistemas operacionais.
- Analisar de forma sistemática os principais componentes de um sistema operacional convencional:
 - gerência de processador,
 - gerência de memória, gerência de entrada e saída e sistemas de arquivos.
- Apresentar como estudo de caso dois dos mais populares sistemas operacionais.

Ementa da Disciplina

- Conceitos básicos de sistemas operacionais: processos, organizações de sistemas operacionais, chamadas de sistema.
- Gerência do processador : estados de processo, escalonamento.
- Entrada e saída: dispositivos e controladores, software de E/S, interrupções, dependência e independência.
- Gerência de memória: partições fixas e variáveis, segmentação, memória virtual. Gerência de arquivos.
- Estudo de Casos

Referências Bibliográfica

R. Oliveira, A. Carissimi, S. Toscani. Sistemas Operacionais, 3ª Edição, Sagra-Luzzato, 2004.

A. Silberchartz, P. Galvin, G. Gagne. Fundamentos de Sistemas Operacionais, 6ª Edição, LTC, 2004.



Referências Bibliográfica

A. S. Tanenbaum. Sistemas Operacionais Modernos, 2ª Edição, Prentice-Hall, 2003.

F. Machado, L. Maia. Arquitetura de Sistemas Operacionais, 4ª Edição, LTC, 2007.



Capítulo 1

Introdução

Introdução

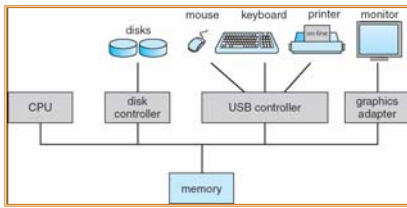
Conteúdo

- O que é um sistema operacional
- Histórico
- Funcionalidades básicas
- Estrutura básica
- Evolução e classificação

Sistema Computacional

□ Cenário Atual

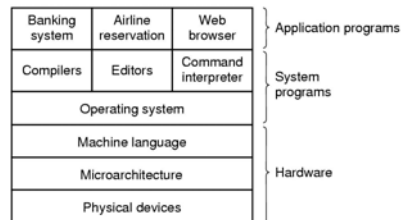
- Uma ou mais CPUs, controladores de devices conectados via uma barramento comum, acessando memórias compartilhadas.
- Execução concorrente de CPUs e devices competindo por recursos.



Introdução

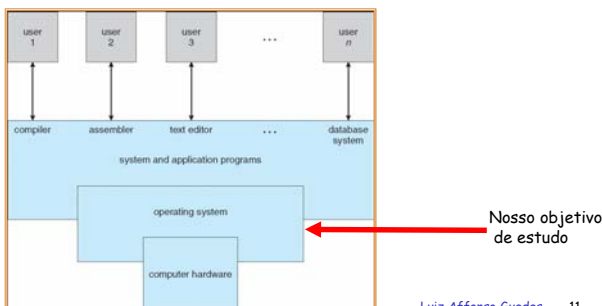
O que é um Sistema Operacional?

- Programa que atua como um intermediário entre um usuário do computador ou um programa e o hardware.



Introdução

Os 4 componentes de sistema computacional



Partida do Computador - Startup

- O programa bootstrap program é carregado na iniciação ou reboot
 - Tipicamente é armazenado na ROM ou EPROM, geralmente conhecidos como **firmware**
 - Inicializa todos os aspectos do sistema
 - Carrega o núcleo do sistema operacional e parte a execução do computador

Introdução

Objetivos de sistema operacional

- ❑ Executar programas de forma conveniente para o usuário.
- ❑ Gerenciar os recursos de software e hardware como um todo.
- ❑ Utilizar os recursos de hardware de forma eficiente e segura.

Introdução

Visões do sistema operacional

- ❑ Como alocador de recursos
 - Gerenciar todos os recursos
 - Resolver conflitos e concorrências por recursos de modo a melhorar a eficiência do sistema como um todo.
- ❑ Como Programa de Controle
 - Controlar a execução de programas para prevenir erros e melhorar o uso do computador.

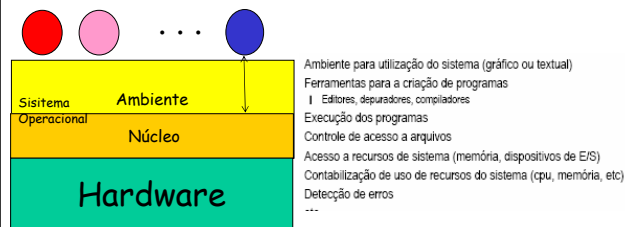
Introdução

Visões do sistema operacional

- ❑ Visão do usuário
 - Interfaces de acesso a recursos e programas.
- ❑ Visão de projeto
 - Organização interna.
 - Mecanismos empregados para gerência de recursos.

Introdução

Estrutura e funcionalidades básicas



Introdução

Núcleo do Sistema Operacional

- ❑ Define a abstração de uma máquina virtual sobre uma máquina real
 - Gerência do processador
 - Gerência de memória real e virtual
 - Gerência de E/S
- ❑ É acessado via chamadas ao sistema (system calls)

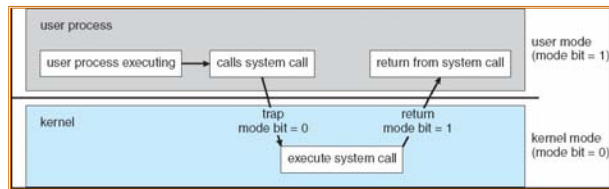
Introdução

Programa de Sistema

- ❑ Programa executados fora do núcleo e compõe o ambiente.
 - Compiladores, ligadores, bibliotecas
 - Interpretadores de comandos (shell, no Unix)
- ❑ Funciona como interface para acesso ao núcleo.

Introdução

Interação entre núcleo ambiente



Exemplos de Chamadas do Sistema

UNIX	Win32	Description
fork	CreateProcess	Create a new process
waitpid	WaitForSingleObject	Can wait for a process to exit
execve	(none)	CreateProcess = fork + execve
exit	ExitProcess	Terminate execution
open	CreateFile	Create a file or open an existing file
close	CloseHandle	Close a file
read	ReadFile	Read data from a file
write	WriteFile	Write data to a file
lseek	SetFilePointer	Move the file pointer
stat	GetFileAttributesEx	Get various file attributes
mkdir	CreateDirectory	Create a new directory
rmdir	RemoveDirectory	Remove an empty directory
link	(none)	Win32 does not support links
unlink	DeleteFile	Destroy an existing file
mount	(none)	Win32 does not support mount
umount	(none)	Win32 does not support mount
chdir	SetCurrentDirectory	Change the current working directory
chmod	(none)	Win32 does not support security (although NT does)
kill	(none)	Win32 does not support signals
time	GetLocalTime	Get the current time

Histórico dos Sistemas Operacionais

- No início:
 - Não havia sistemas operacionais
 - O usuário era o operador e programador da máquina
- Necessidade da evolução
 - Melhor utilização de recursos
 - Avanço tecnológico
 - Incorporação de novos serviços
- A evolução dos SOs está associada com a própria evolução da informática.

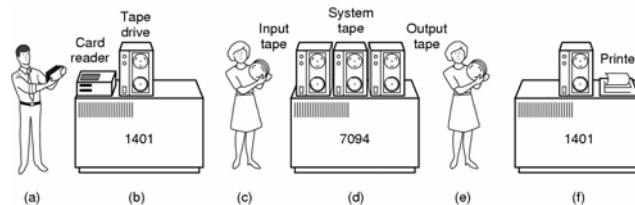
Evolução dos Computadores

- Primeira geração 1945 - 1955
 - Válvulas, cabos de ligação
- Segunda geração 1955 - 1965
 - transistores, sistemas em lote (batch)
- Terceira geração 1965 - 1980
 - ICs e multiprogramação
- Quarta geração 1980 - present
 - Computadores pessoais

Históricos - Sistemas em Lote (Batch)

- Introdução de operadores profissionais
- Criação do conceito de Job
 - Programas e serem compilados e executados.
 - Eram organizados em lotes
 - Dados e código eram agrupados, não havendo interação com o usuário.
 - Execução seqüencial de lotes
 - Passagem entre Jobs é manual

Histórico - Sistemas em Lote (Batch)

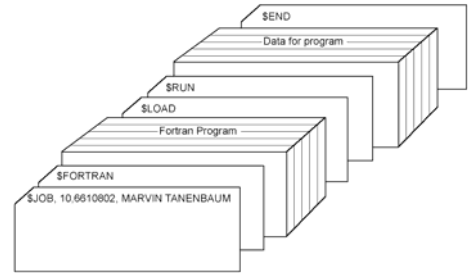


- Seqüência manual
 - Leitura
 - Execução
 - Impressão

Histórico - Monitor Residente

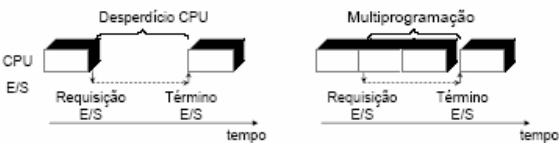
- ❑ Sequenciamento automático de Jobs
 - Um Job passa o controle para outro de forma automática.
 - Sistema Operacional rudimentar.
 - Monitor Residente
 - Programa que ficava residente na memória
 - Primeiro programa a ser executado
 - Passava o controle ao Job.
 - Quando um Job terminava, o controle voltava ao Monitor
 - Centralizava o acesso a periféricos
 - Apenas um programa por vez

Histórico - Monitor Residente



Histórico - Sistema Batch Multiprogramados (Multiprogramação)

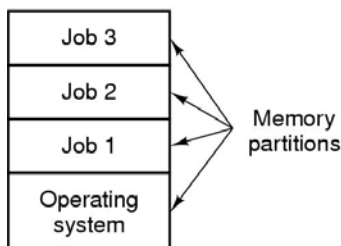
- ❑ Operações de E/S são lentas → uma programa por vez é uma estratégia simples, mas ineficiente.
- ❑ Necessidade de multiprogramação
 - Gerenciamento mais complexo



Multiprogramação

- ❑ Possibilidade de mais de um programa em execução simultaneamente
 - Necessidade de interrupção
 - Mecanismos mais eficientes de acesso a disco
- ❑ Necessidade de sistemas operacionais mais complexos
- ❑ Sistemas timesharing
 - Terminais de usuários
 - Máquinas virtuais

Multiprogramação



Sistema Monotarefa e Multitarefa

- ❑ Monotarefa
 - Uma tarefa por vez (simplicidade)
 - MS-DOS
- ❑ Multitarefa
 - Mais de uma tarefa simultaneamente
 - Não preemptivo
 - Windows de 16 bits.
 - Preemptivo (mais complexos e eficientes)
 - Unix, Windows NT, Windows 2000, etc.

Sistema Monousuários e Multiusuários

- Monousuário
 - Apenas um usuário por vez, mais por ser multitarefa
- Multiusuário
 - Multiterminais
 - Sistemas em rede
 - Sistemas Distribuídos
 - Unix, Windows NT, Windows 2000, etc.

Sistemas Operacionais Distribuídos

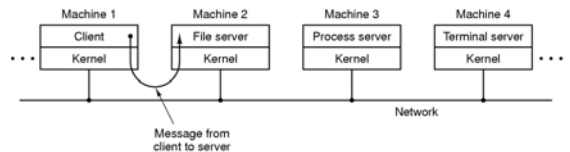
- A idéia básica é distribuir as atividades entre diversos computadores
- Essa distribuição deve ser transparente par ao usuário / aplicação
 - Deve-se imaginar idealmente que haja apenas uma máquina
- Vantagens
 - Compartilhamento de recursos
 - Balanceamento de carga
 - Aumento da confiabilidade
- Desvantagem
 - Sistema mais complexos

Sistemas Operacionais Distribuídos

- Sistemas fracamente acoplados
 - Cada máquina possui o seu próprio sistema operacional
 - Comunicação por troca de mensagens
- Sistemas fortemente acoplados
 - Sistemas paralelos
 - Auto desempenho
 - Comunicação por memória compartilhada

Sistemas Operacionais de Rede

- Paradigma Cliente-servidor
 - Distribuir os serviços do SO entre diversos computadores
 - Não há transparência de localização



Outros Tipos de Sistemas Operacionais

- Sistemas Operacionais de Tempo-real
 - Aplicações que exigem garantias determinísticos de tempo
 - Controle de processos industriais
 - Tempo-real soft e tempo-real hard
- Sistemas Operacionais Embarcados
 - Altamente compactos
 - Smart cards
- SO Distribuídos, de tempo-real e embarcados, seguros, ...