

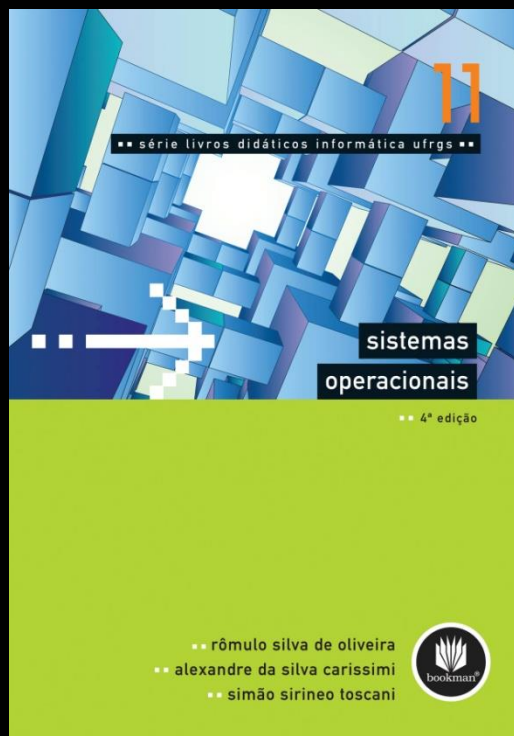
■ ■ série de livros didáticos informática ufrgs

11



**bookman**<sup>®</sup>  
EMPRESA DO GRUPO ARTMED  
[www.bookman.com.br](http://www.bookman.com.br)

**.inf**  
INSTITUTO  
DE INFORMÁTICA  
UFRGS

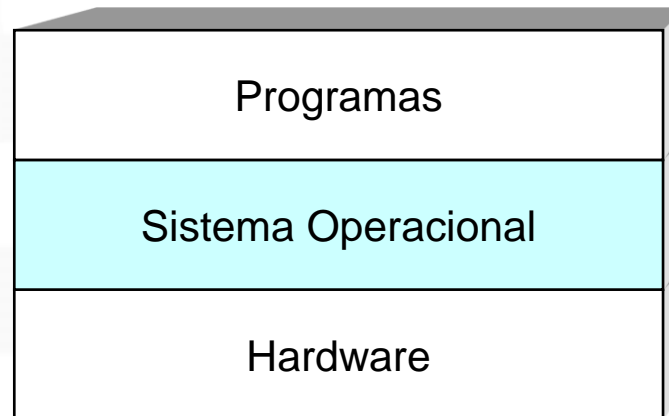


## Sistemas Operacionais

Rômulo Silva de Oliveira  
Alexandre da Silva Carissimi  
Simão Sirineo Toscani

# Introdução

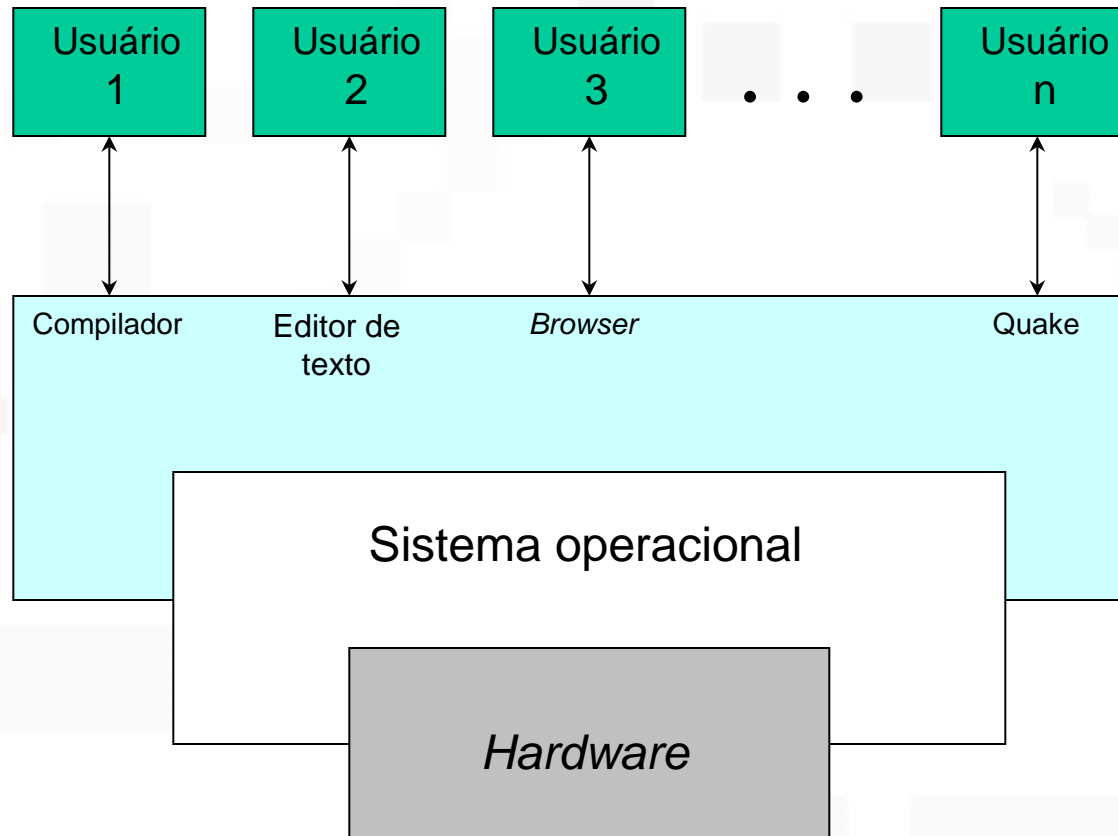
- O que se espera de um sistema de computação?
  - Execução de programas de usuários
  - Permitir a solução de problemas
- Sistema operacional é um programa colocado entre o hardware do computador e os programas dos usuários de forma a atingir esses dois objetivos



# Componentes genéricos de um sistema computacional (1)

- Hardware
  - Recursos básicos (memória, dispositivos de E/S, CPU).
- Sistema operacional
  - Controla e coordena o uso do hardware entre vários programas aplicativos e usuários.
- Programas aplicativos e de sistema
  - Define como os recursos de hardware são empregados na solução de um problema (compiladores, jogos, banco de dados,...)
- Usuários
  - Pessoas, máquinas, outros computadores,...

# Componentes genéricos de um sistema computacional (2)



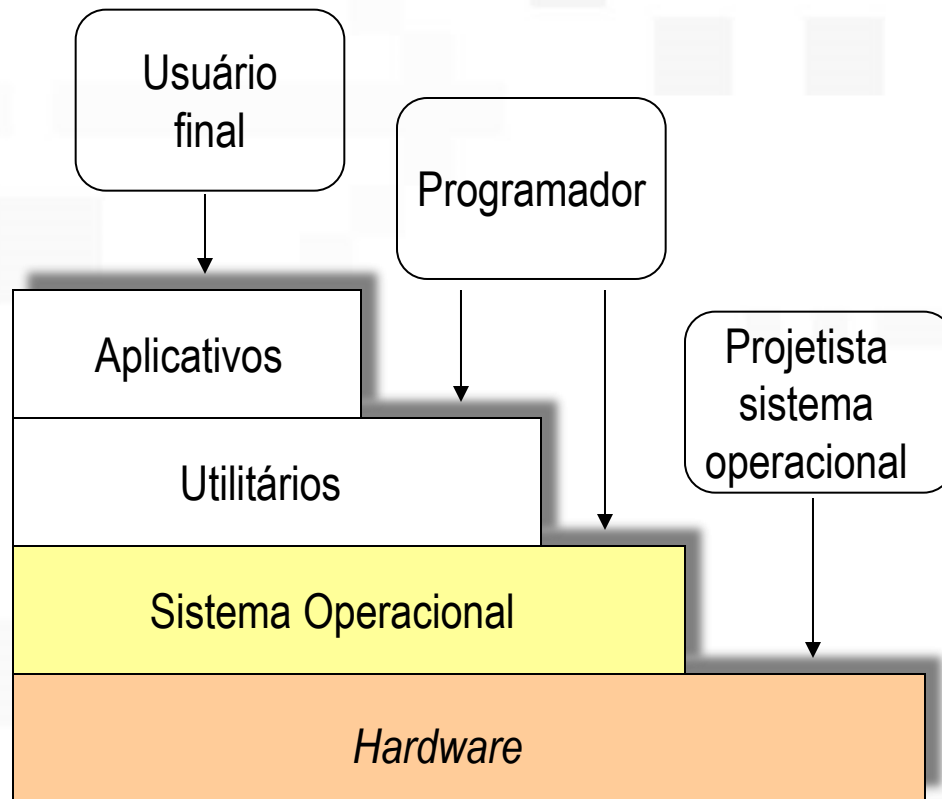
# Sistema operacional: conceito

- Sistema operacional
  - Um programa que controla a execução de programas aplicativos
  - Interface entre aplicativos e o hardware
- Duas formas de ver um sistema operacional:
  - Alocador de recursos
  - Programa de controle

# Objetivos do sistema operacional

- Tornar mais conveniente a utilização de um computador
  - “Esconder” detalhes internos de funcionamento
- Tornar mais eficiente a utilização de um computador
  - Gerenciamento “justo” dos recursos do sistema
- Facilitar a evolução do sistema (desenvolvimento, teste e atualização de novas facilidades)

# Sistema operacional: interface entre usuário/computador



# Serviços oferecidos pelo sistema operacional (1)

- Criação de programas
  - Editores, depuradores, compiladores
- Execução dos programas
  - Carga de programas em memória
- Acesso a dispositivos de E/S
- Controle de acesso a arquivos
- Acesso a recursos de sistema
  - Proteção entre usuários



# Serviços oferecidos pelo sistema operacional (2)

- Contabilidade
  - Estatísticas
  - Monitoração de desempenho
  - Sinalizar *upgrades* necessários *hardware* (memória, disco, etc)
  - Tarifação de usuários
- Detecção de erros
  - Erros de *hardware*
    - e.g.: erros de memória, falha em dispositivos de E/S, etc...
  - Erros de programação
    - e.g.: *overflow*, acesso não autorizado a posições de memória, etc...
  - Aplicação solicita recursos que o sistema operacional não pode alocar (segurança, falta do recurso, etc)

# As diferentes “imagens” de um sistema operacional

- Sistema operacional na visão do usuário:
  - Imagem que um usuário tem do sistema
  - Interface oferecida ao usuário para ter acesso a recursos do sistema
    - Chamadas de sistema
    - Programas de sistema
- Sistema operacional na visão de projeto
  - Organização interna do sistema operacional
  - Mecanismos empregados para gerenciar recursos do sistema

# Chamadas de sistema

- Forma que programas solicitam serviços ao sistema operacional
  - Análogo a sub-rotinas
    - Transferem controle para o sistema operacional invés de transferir para outro ponto do programa
- É o núcleo (*kernel*) do sistema operacional que implementa as chamadas de sistema
  - Existem chamadas de sistema associadas a gerência do processador, de memória, arquivos e de entrada/saída
- Variação: *micro-kernel*
  - Serviços básicos são implementados pelo micro-kernel
  - kernel implementa demais serviços empregando esses serviços básicos

# Programas de sistema

- Programas executados fora do *kernel* (utilitários)
- Implementam tarefas básicas:
  - Muitas vezes confundidos com o próprio sistema operacional
    - e.g. compiladores, *assemblers*, ligadores, etc.
  - Interpretador de comandos
    - Ativado sempre que o sistema operacional inicia uma sessão de trabalho
      - e.g.: bash, tsch, sh, etc...
    - Interface gráfico de usuário (GUI)
      - e.g.: Família windows, MacOs, etc...

# Histórico de sistemas operacionais

- Primórdios:
  - Sistema operacional inexistente
  - Usuário é o programador e o operador da máquina
  - Alocação do recurso “computador” feito por planilha
- Evolução foi motivada por:
  - Melhor utilização de recursos
  - Avanços tecnológicos (novos tipos de hardware)
  - Adição de novos serviços

# Sistemas em lote (*batch*)

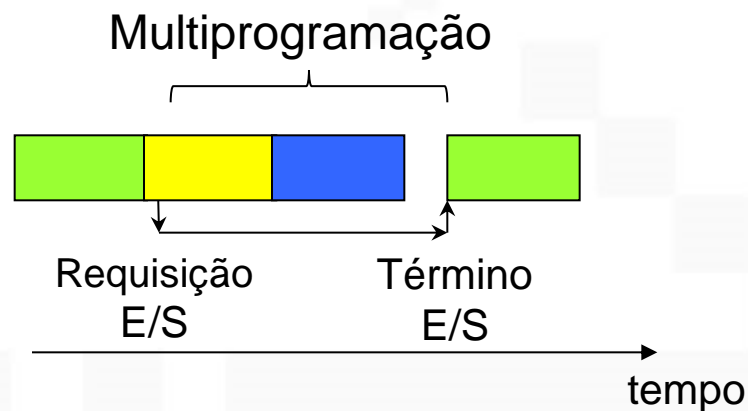
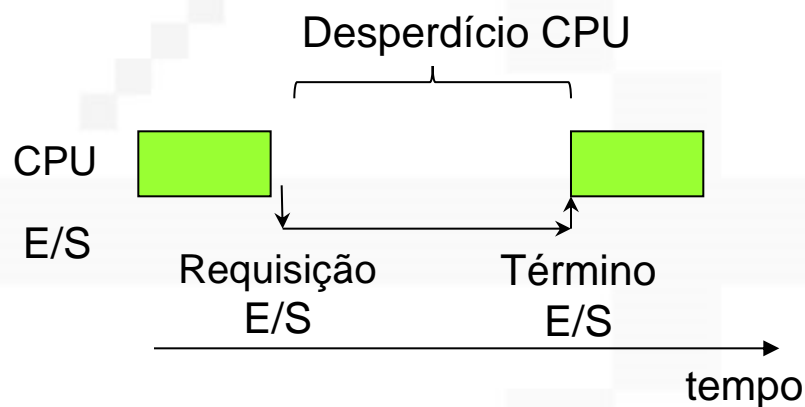
- Introdução de operadores profissionais
  - Usuário não era mais o operador da máquina
- *Job*
  - Programa a ser compilado e executado, acompanhado dos dados de execução (cartões perfurados)
  - Jobs são organizados em lote (*batch*)
    - Necessidades semelhantes (e.g. mesmo compilador)
- Passagem entre diferentes *jobs* continua sendo manual

# Monitor residente

- Evolução:
  - Sequenciamento automático de *jobs*, transferindo o controle de um *job* a outro
  - Primeiro sistema operacional (rudimentar)
- Monitor residente:
  - Programa que fica permanentemente em memória
  - Execução inicial
  - Controle é transferido para o *job*
    - Cartões de controle
  - Quando o *job* termina, o controle retorna ao monitor
  - Centraliza as rotinas de acesso a periféricos disponibilizando aos programas de usuário

# Sistema *batch* multiprogramados (multitarefa)

- Monitor residente permite a execução de apenas um programa a cada vez
- Desperdício de tempo de CPU com operações de E/S
- Evolução:
  - Manter diversos programas na memória ao mesmo tempo
  - Enquanto um programa realiza E/S, outro pode ser executado





# Multiprogramação

- Manter mais de um programa em “execução” simultaneamente
- Duas inovações de *hardware* possibilitaram o surgimento da multiprogramação
  - Interrupções
    - Sinalização de eventos
  - Discos magnéticos
    - Acesso randômico a diferentes *jobs* (programas) no disco
    - Melhor desempenho em acessos de leitura e escrita

# Sistemas *timesharing*

- Tipo de multiprogramação
- Usuários possuem um terminal
  - Interação com o programa em execução
- Ilusão de possuir a máquina dedicada a execução de seu programa
  - Divisão do tempo de processamento entre usuários
  - Tempo de resposta é importante

# Sistemas monousuário e multiusuário

- Sistemas monousuário
  - Projetados para serem usados por um único usuário
    - e.g.: MS-DOS, Windows 3.x, Windows 9x, Millenium
- Sistemas multiusuário
  - Projetados para suportar várias sessões de usuários em um computador
    - e.g.: Windows NT (2000), UNIX

# Sistemas multitarefa e monotarefa

- Sistemas monotarefa
  - Capazes de executar apenas uma tarefa de cada vez
    - e.g.: MS-DOS
- Sistemas multitarefas:
  - Capazes de executar várias tarefas simultaneamente
  - Existem dois tipos de sistemas multitarefa:
    - Não preemptivo (cooperativo)
      - e.g.: Windows 3.x, Windows9x (aplicativos 16 bits)
    - Preemptivo
      - e.g.: Windows NT, OS/2, UNIX, Windows9x (aplicativos 32 bits)

# Sistemas distribuídos (1)

- Distribuir a realização de uma tarefa entre vários computadores
  - Sistema distribuído:
    - Conjunto de computadores autônomos interconectados de forma a possibilitar a execução de um serviço
    - Existência de várias máquinas é transparente
    - Software fornece uma visão única do sistema
    - Palavra-chave: transparência
- « *A distributed system is one in which the failure of a computer you didn't even know existed can render your own computer unusable* » Leslie Lamport.

# Sistemas distribuídos (2)

- Sistemas fracamente acoplados (*loosely coupled system*)
  - Máquinas independentes
    - Cada máquina possui seu próprio sistema operacional
  - Comunicação é feita através de troca de mensagens entre processos
- Vantagens
  - Compartilhamento de recursos
  - Balanceamento de carga
  - Aumento da confiabilidade

# Sistemas paralelos (1)

- Máquinas multiprocessadoras possuem mais de um processador
- Sistemas fortemente acoplados (*tightly coupled system*)
  - Processadores compartilham memória e relógio comuns
  - Comunicação é realizada através da memória
- Vantagens:
  - Aumento de *throughput* (número de processos executados)
  - Aspectos econômicos
  - Aumento de confiabilidade
    - *Graceful degradation*
    - Sistemas *fail-soft*

# Sistemas paralelos (2)

- *Symmetric multiprocessing (SMP)*
  - Cada processador executa uma cópia idêntica do sistema operacional
  - Vários processos podem ser executados em paralelo sem perda de desempenho para o sistema
  - A maioria dos sistemas operacionais atuais suportam SMP através do conceito de *multithreading*
- *Asymmetric multiprocessing*
  - Cada processador executa uma tarefa específica:
    - e.g.: processador mestre para alocação de tarefas a escravos



# Sistemas de tempo real

- Empregado para o controle de procedimentos que devem responder dentro de um certo intervalo de tempo
  - e.g.: experimentos científicos, tratamento de imagens médicas, controle de processos, etc.
- Noção de tempo real é dependente da aplicação
  - Milisegundos, minuto, horas, etc.
- Dois tipos:
  - *Hard real time*
    - Tarefas críticas são completadas dentro de um intervalo de tempo
  - *Soft real time*
    - Tarefa crítica tem maior prioridade que as demais

# Leituras complementares

- R. Oliveira, A. Carissimi, S. Toscani; Sistemas Operacionais. Editora Bookman, 2010.
  - Capítulo 1
- A. Silberchatz, P. Galvin, P. Gagne; Applied Operating System Concepts. Addison-Wesley, 2000, (1ª edição).
  - Capítulo 1
- W. Stallings; Operating Systems. (4<sup>th</sup> edition). Prentice Hall, 2001.
  - Capítulo 2