

SISTEMAS OPERATIVOS

[HTTP://MOODLE.DSI.UMINHO.PT/MOODLE](http://moodle.dsi.uminho.pt/moodle)
(SOLTSI2011)

Helena Rodrigues
Departamento de Sistemas de Informação
Escola de Engenharia
Universidade do Minho

Introdução aos Sistemas Operativos

Licenciatura em Tecnologias e Sistemas de Informação

Sistemas Operativos

Helena Rodrigues
(helena@dsi.uminho.pt)

Introdução aos Sistemas Operativos

Bibliografia:

Silberschatz, A., Galvin, P., Gagne, Greg, *Applied Operating Systems Concepts*, Addison Wesley, 2000. (1ª edição: cap. 1, cap.3 p. 45-54; 7ª edição: cap. 1, cap.2 p. 39-56)

José Alves Marques, Paulo Ferreira, Carlos Ribeiro, Luís Veiga e Rodrigo Rodrigues, *Sistemas Operativos*, FCA, 2009, cap. 1 e cap. 2

Introdução aos Sistemas Operativos

Resultados de aprendizagem:

- Explicar as funções de um sistema operativo
- Explicar a diferença entre o sistema operativo e outros programas do sistema
- Descrever as diferentes classes de sistemas operativos
- Descrever a organização de um sistema operativo
- Descrever as componentes e serviços de um sistema operativo
- Descrever a evolução dos sistemas operativos até aos sistemas operativos modernos
- Explicar as funções de um sistema operativo face a evolução dos mesmos até aos sistemas operativos modernos
- Explicar a função de um interpretador de comandos

Introdução aos Sistemas Operativos

Questões:

1. Quais as componentes de um sistema de computação?
2. Que “tipos” de aplicações existem num sistema de computação

Componentes de um Sistema de Computação

1. *Hardware* – oferece os recursos de computação básicos (CPU, memória, dispositivos de I/O).
2. Programas do sistema (incluindo o Sistema Operativo) – controlam e coordenam a utilização do hardware pelos vários programas de aplicação de diferentes utilizadores.
3. Programas de aplicação – definem uma forma de utilização de recursos para resolver os diferentes problemas dos utilizadores (compiladores, bases de dados, jogos, programas de gestão).
4. Utilizadores (pessoas, outros computadores, ...).

Introdução aos Sistemas Operativos

Questões:

3. Vamos supor que eliminamos todos os programas da classe "Programas do sistema".

3.1 Quais as consequências para quem desenvolve programas de aplicação?

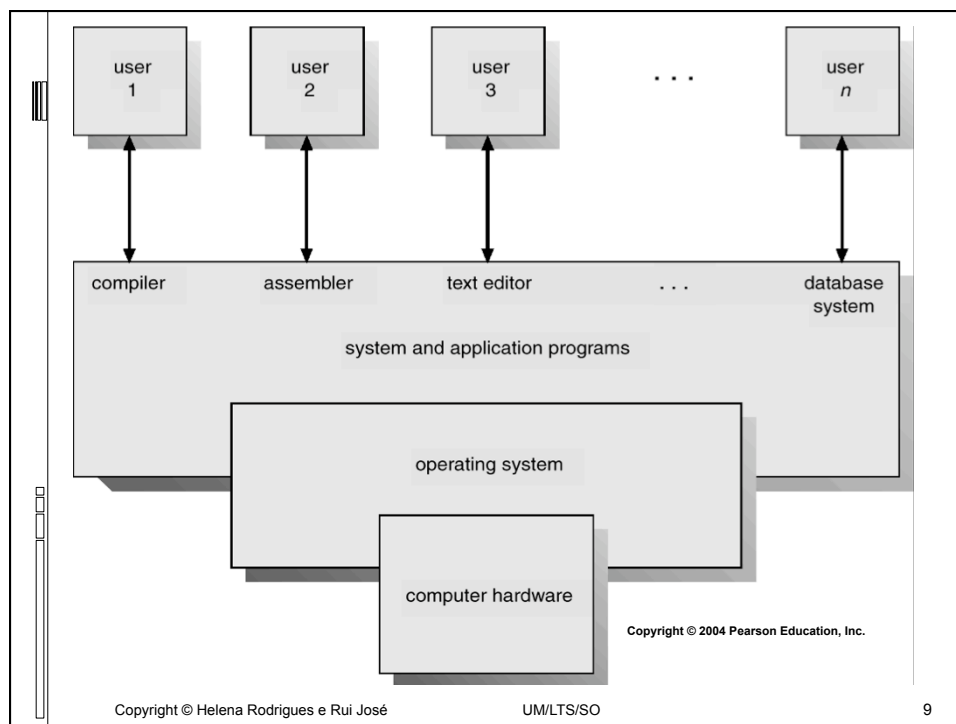
3.2 Nas condições anteriores, imaginem que estão a desenvolver um programa e necessitam de aceder ao disco? Quais os passos a executar?

Para que serve um S.O

- O hardware por si só não é particularmente fácil de utilizar
- Programas precisam de muitas funcionalidades comuns e, caso estejam em execução simultânea, de um entidade que coordene a utilização de recursos.
- Essas funcionalidades são colocadas num programa especial designado Sistema Operativo que irá facilitar o desenvolvimento e execução dos outros programas

O que é um Sistema Operativo?

- Um programa que age como um intermediário entre o utilizador e o hardware criando um ambiente adequado para que um utilizador possa executar os seus programas
- Um dos objectivos de um SO é fazer com que um sistema de computação seja mais fácil de utilizar (*convenient*)
 - Visão máquina estendida
- Outro objectivo é promover uma utilização eficiente dos recursos de hardware
 - Visão gestor de recursos



Sistemas Operativos

Multi-programação

Multi-tarefa

Multi-utilizador

Multi-processorador

De Rede

Distribuídos

De tempo-real



Introdução aos Sistemas Operativos

Questões:

4. Quais as principais limitações dos sistemas *batch* iniciais no que diz respeito à utilização do CPU? Que soluções sugere para ultrapassar essas limitações?

Sistemas *Batch*

- No início:
 - O sistema era manuseado apenas pelo operador
 - Equipado com um leitor de cartões
 - Utilizador prepara uma tarefa (programa, dados e informação de controlo) para submissão
 - Operador controla execução das várias tarefas
 - Resultado da execução entregue posteriormente ao utilizador
- Problema: baixo desempenho
 - I/O e processamento não se sobrepõem; leitor de cartões muito lento.

Introdução aos Sistemas Operativos

Questões:

5. Quais os principais requisitos para o suporte a sistemas de multi-programação?

Sistemas Multi-programação

- Várias tarefas são mantidas em memória ao mesmo tempo, sendo o CPU multiplexado entre elas.

Questão 5. Quais os principais requisitos para o suporte a sistemas de multi-programação?

- Requisitos:
 - Gestão de memória - o sistema tem de atribuir a memória às diferentes tarefas.
 - Escalonamento do CPU - o sistema tem de seleccionar a tarefa a executar de entre todas as tarefas prontas a executar.
 - Protecção entre tarefas
 - suporte a I/O.

Sistemas *Time-Sharing*

- *Time-Sharing ou multitasking*
 - fornece um serviço *on-line*;
 - cada utilizador dispõe de um terminal
 - CPU é multiplexado entre várias tarefas
 - Trocas entre tarefas muito rápidas
 - É imposta a divisão do tempo disponível do processador pelos vários utilizadores, atribuindo-lhes fatias de tempo que permitem dar-lhes a sensação de dispor de uma máquina dedicada.

Sistemas *Time-Sharing*

- Requisitos
 - gestão de memória
 - gestão de ficheiros
 - sistemas de protecção
 - algoritmos de escalonamento de tempo de CPU sofisticados
 - e mais.

Sistemas para PC

- no passado apresentavam requisitos diferentes das grandes máquinas
- a evolução de sistemas complexos de janelas introduziu o requisito da multi-tarefa
- a participação em redes de computadores fez evoluir o requisito de protecção de ficheiros (multi-utilizador)
- versões modernas incluem o Windows XP e o Linux

Sistemas Paralelos

- Sistemas multi-processador em que os vários processadores trabalham em estreita colaboração
- *Symmetric multiprocessing (SMP)*
 - Processadores funcionam em modelo *peer*
- *Asymmetric multiprocessing*
 - Processadores funcionam em modelo *master-slave*

Sistemas Operativos Distribuídos

- Distribui a computação por vários computadores
 - *Loosely coupled system* – cada computador tem a sua memória local; computadores comunicam através de linhas de comunicação.
 - Vantagens dos sistemas distribuídos
 - partilha de recursos
 - Aumento de desempenho e distribuição de carga
 - Robustez
 - Comunicação

Sistemas Operativos Distribuídos

- Sistemas Operativos de rede
 - partilha de ficheiros
 - facilidades de comunicação
 - independente de outros computadores na rede
- Sistema Operativo Distribuído
 - menos autonomia entre computadores
 - dá a impressão de que existe apenas um sistema operativo a controlar um conjunto de computadores em rede.

Sistemas de tempo real

- Sistemas utilizados quando existem requisitos de tempo que (devem) têm de ser garantidos.
- Um resultado só é válido se for correcto e se for produzido dentro dos limites de tempo definidos
- Sistemas *hard real-time*
 - Garantem que as tarefas críticas serão terminadas no tempo especificado
 - Incompatível com time-sharing
 - Ex. Sist. travagem de um carro

Sistemas de tempo real

- Sistemas *soft real-time*
 - Permite especificar restrições de tempo e gere as prioridades de forma a tentar cumprir com esses limites
 - Compatível com sistemas time-sharing
 - Ex. Transmissão de vídeo

Organização do Sistema Operativo

- Componentes típicos de um SO
- Serviços de um SO
- Chamadas ao Sistema (*System Calls*)
- Programas do Sistema

Introdução aos Sistemas Operativos

Questões:

- 6.1. O que é um processo?
- 6.2. Que informação o sistema operativo precisa manter sobre um processo?

Componentes

- **Gestão de Processos**
 - Um processo é um programa em execução. Um processo requer um determinado número de recursos, incluindo tempo de CPU, memória, ficheiros, e dispositivos de I/O.
 - programa em execução inclui:
 - código, dados e *stack* (espaço de endereçamento)
 - *program counter*, *stack pointer* e outros registos
 - estado da execução
 - recursos atribuídos

Componentes

- **Gestão de Processos**
 - O sistema operativo, através do gestor de processos, é responsável por:
 - Criar e terminar processos
 - Suspende e acordar processos
 - Oferecer mecanismos para:
 - sincronização de processos
 - comunicação entre processos
 - identificação de processos em ambientes de multiprogramação (uid - *user identification* - e gid - *group identification*)

Introdução aos Sistemas Operativos

Questões:

- 7.1. Onde é mantido (componente de hardware) um programa quando não está em execução?
- 7.2. Onde é mantido (componente de hardware) um programa que está em execução?
- 7.3. O que representam os endereços de um programa em *assembly*?
- 7.4. O que significa a expressão "Não há memória suficiente"?
- 7.5 O que significa termos dois programas em execução?



Componentes

- Gestão de Memória
 - A memória principal é um enorme *array* de *bytes* cada um com o seu endereço. É acedida directamente pelo CPU e dispositivos de I/O
 - Os programas para serem executados tem de ser carregados, pelo menos parcialmente, para memória principal.



Componentes

- Gestão de Memória
 - O sistema operativo, através do gestor de memória, é responsável por:
 - Manter o registo de quais as partes de memória estão a ser usados num dado instante e por quem.
 - Decidir sobre que processos carregar para memória quando existe espaço livre.
 - Atribuir e libertar espaço de memória quando necessário.

Introdução aos Sistemas Operativos

Questões:

- 8.1. O que é um ficheiro? Para que serve? Porque é necessário?
- 8.2. Que operações deverá o sistema operativo oferecer no contexto de manuseamento de ficheiros?

Componentes

- **Gestão de ficheiros**
 - memória principal é volátil e diminuta para armazenar todos os programas e dados permanentemente
 - os sistemas de computação oferecem a memória secundária.
 - Um **ficheiro** é uma estrutura de dados que esconde a complexidade dos dispositivos físicos de armazenamento de informação (discos, ...);
 - Tipos de ficheiros: ficheiros de dados ou programas executáveis, directorias, ficheiros especiais

Componentes

- **Gestão de Ficheiros**
 - O sistema operativo, através do sistema de ficheiros, é responsável por:
 - Criar e apagar ficheiros
 - Criar e apagar directorias
 - Suportar primitivas para manipular ficheiros e directorias
 - Associar ficheiros ao sistema de memória secundária
 - Gestão do espaço livre em disco
 - Atribuição de espaço em disco
 - Escalonamento de serviço de I/O

Introdução aos Sistemas Operativos

Questões:

- 9.1. Em relação às operações para acesso ao disco descritas no início da aula, qual é a interface de mais alto nível?
- 9.2. Se a situação fosse referente à unidade de CD-ROM, que operações teriam de ser alteradas?
- 9.3. E se a situação fosse referente à placa de rede?
- 9.4. E à impressora?

Componentes

- O sistema operativo, através do **gestor de I/O**, é responsável por:
 - esconder a complexidade dos dispositivos de I/O ao utilizador, oferecendo-lhe uma interface abstracta de alto nível
 - Conceitos básicos
 - interface para *device-drivers* genérica (componente independente do dispositivo)
 - *Drivers* específicos (componente dependente do dispositivo)

Componentes

- Sistema de protecção
 - o mecanismo de protecção é responsável por controlar o acesso de processos ou utilizadores tanto a recursos do sistema como do utilizador.
 - O sistema operativo, através do sistema de protecção, é responsável por:
 - distinguir entre o acesso autorizado e o acesso não autorizado.
 - Especificar o controlo a impor.
 - Providenciar uma forma de implementar essa protecção.

Componentes

- O interpretador de comandos
 - programa que estabelece uma interface entre o utilizador e o sistema operativo para a execução de operações genéricas
 - interpreta e executa comandos ao sistema operativo, como por exemplo executar um programa
 - implementa um ciclo de leitura, interpretação e execução de comando
 - No *Unix/Linux* é conhecida pela *shell*

Chamadas ao sistema

- Chamadas ao sistema (*system calls*)
 - oferecem uma interface entre os programas e o sistema operativo.
 - Genericamente, disponíveis como instruções *assembly*.
 - C, C++ e outras linguagens permitem fazer chamadas ao sistema directamente.
 - Podem ser agrupadas por serviços
 - Gestão de ficheiros
 - Gestão de memória
 - Gestão de processos
 -

Questões

- Será necessário um sistema operativo para executar um programa em código máquina?
- Fará sentido imaginar-se um sistema de computação sem sistema operativo? Em que condições é que isso poderá fazer sentido?

Questões

- Quais são as principais vantagens de suportar multi-programação num sistema de computação
- Quais deverão ser os limites do Sistema Operativo? Considere as vantagens e desvantagens de incluir no sistema operativo:
 - Um interpretador de comandos
 - Um cliente de mail
 - Um browser web