

Sistemas Operativos

LEI + LCC

Grupo de Sistemas Distribuídos
<http://gsd.di.uminho.pt>

.1

As más notícias

- Não basta ir ao Google ou “ver” os exercícios/testes
- É uma UC de engenharia:
 - É preciso usar a “massa cinzenta”
 - Também é preciso a “sujar as mãos”
=> programar, testar, configurar, debug...
- Muitos alunos encravam em deficiências passadas, por exemplo na dificuldade de raciocinar, concepção de algoritmos, C, saber como funciona um computador...
=> BATER CÓDIGO!

.2

As boas notícias

- Apesar da “mística” que rodeia os sistemas operativos...
- basta um pouco de trabalho* e bom senso para fazer esta unidade curricular
- e ultrapassar grande parte das deficiências anteriores!

.3

(*) O trabalho inclui assistir às aulas e mais 2 a 3 horas/semana

Equipa Docente

LSDS @ HASLab (Large Scale Distributed Systems)

- Responsável + aulas teóricas + 1 turno PL
 - fsm@di.uminho.pt
- Aulas práticas
 - cbm, jop, rco, vff@di
- Horário de atendimento a definir

.4

Objectivos

Ajudar a perceber como funcionam os computadores

- Em termos físicos, o que é uma aplicação informática?
- Que recursos necessita?
- Como devem ser geridos esses recursos?
- A aplicação interage com outras? (i.e. **Concorrência**)
- Isto está lento... Que fazer?
- Foi abaixo... Perdi tudo?

.5

Bibliografia recomendada

- Silberschatz, Galvin and Gagne, *Operating System Concepts*, John Wiley & Sons, 2010.
- Carlos Ribeiro, Alves Marques, 2^a-ed, FCA Editora Informática, 2012.

-> **FSM 2004, VOU FAZER SISTEMAS OPERATIVOS**

.7

Programa

- Recapitulação de conceitos de programação de sistemas
- Gestão de processos, memória, ficheiros, periféricos
- Alguma **programação concorrente** (de baixo nível)
- E mãos na massa:
 - Aulas práticas em ambiente Linux
 - Nada de janelas, nem ratos...
 - Terminal com *Bash*, comandos, pipelines...
 - Programação de “baixo nível”: C, syscalls, libs...



.6

Bibliografia Adicional

- Man !!!!!!!
- R. Stevens, *Advanced Programming in the Unix Environment*, Addison Wesley, 1990.
- Google, slashdot...

.8

Slides

- Baseados os originais que acompanham os livros recomendados (em especial Silberschatz)
- (Progressivamente) disponíveis no Blackboard:
- Servem apenas de “âncora” ao estudo;

Não chegam para responder aos testes!!!

.9

Avaliação

- 1/3 **Trabalho Prático** em grupo (<=3 elementos)
- 2/3 **Prova escrita** (teste individual e/ou exame de recurso)
- Há nota mínima no TP (10) e no teste/exame (9)
- É obrigatório ter nota positiva no TP para ser admitido ao teste, exame de recurso e exame especial, se for caso disso.
- Nota do TP em 2014/2015 pode ser “reutilizada” mas é truncada a 15 valores (2013/2014 e anteriores fazem novo trabalho)

.11

Aulas práticas

- Cada aula prática tem um “guião” muito detalhado.
 - Normalmente só duram uma semana
 - Fazem sempre falta para a aula seguinte (+ TP)
 - Sempre que resolver uma alínea, pare e pense:
O que é que EU aprendi com este exercício?
- Recomenda-se vivamente:
 - estudar o guião antes da respectiva aula
 - usar a aula para tirar dúvidas e não para copiar o que está no quadro, no colega do lado, no Google...
 - terminar em casa todas as alíneas não resolvidas durante a aula; se necessário, peça ajuda por mail

.10

Avaliação

- Prova escrita + exame com parte teórica e prática
 - Quase tudo tem a ver com concorrência!
 - Valoriza-se a capacidade de raciocínio e a concepção de algoritmos (por oposição à utilização de “padrões” de soluções)
 - As perguntas insistem sempre nos “porquês”, na justificação, demonstração ou prova. Não basta repetir o que se decorou
- **Ninguém faz a disciplina apenas com a parte teoria**
- **Dificilmente a fará só com a prática**

.12

Programa

- Introdução (à *programação de sistemas*)
- Gestão de processos
- Noções de programação concorrente
- Gestão de memória
- Gestão de periféricos
- Gestão de ficheiros

.13

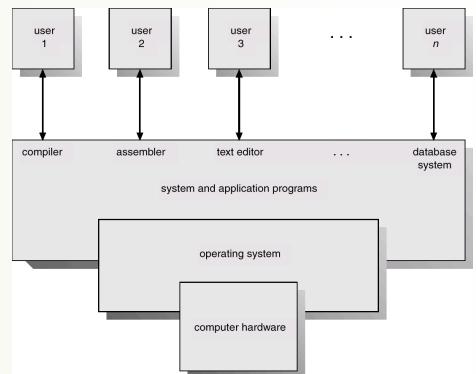
Para que serve um computador?

- Para executar programas (aplicações)
- Que facilitam a vida aos utilizadores

.14

O que é um Sistema Operativo?

- Software que actua como **intermediário** entre os utilizadores e o hardware



.15

Portanto...

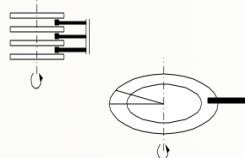
- O Sistema operativo deve colocar o hardware à disposição dos programas e utilizadores, mas de uma forma
 - **conveniente,**
 - **protegida,**
 - **eficiente,**
 - **justa,**
 - ...

.16

O Sistema Operativo...

Pode ser visto como

- Extensão da máquina, simula uma *máquina virtual*
`open()`, `read()`, `write()`...
- Gestor de recursos



.17

Objectivos (1)

• Conveniência

- SO esconde os detalhes do hardware
 - e.g. [dimensão e organização da memória ou disco](#)
- Simula máquina virtual com valor acrescentado
 - e.g. [cada processo executa numa “máquina” extendida e protegida](#)
- Fornece API mais fácil de usar do que o hardware
 - e.g. [Operações sobre ficheiros vs. blocos em disco](#)

.18

Na prática...

- É o Sistema Operativo quem define a “**personalidade**” de um computador
- Como se comporta o mesmo computador (hardware) após ter arrancado
 - MSDOS?
 - Windows 95?
 - Windows 10?
 - Mac OSX?
 - Linux (Ubuntu, Kali, ...)?



.19

Objectivos (2)

• Eficiência

- SO controla a alocação de recursos
 - Se 3 programas usarem a impressora ao mesmo tempo → [sai lixo?](#)
 - Programa em ciclo infinito → [computador bloqueia?](#)
 - Processo corrompe a memória dos outros → [programas morrem?](#)
- Multiplexação:
 - Tempo: processos usam o recurso à vez (impressora, CPU)
 - Espaço: recurso é partilhado simultaneamente por vários processos (memória central, disco)

.20

Objectivos (3)

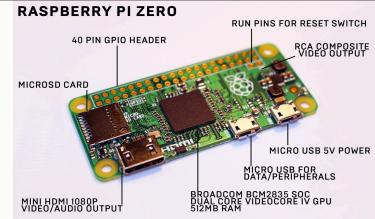
- Recapitulemos então os objectivos gerais de um SO
 - Conveniência
 - Eficiência
- Os nossos critérios de avaliação serão portanto...
 - Dá jeito?
 - É eficiente ou diminui tempos de resposta?
 - Nem uma nem outra?

21

Evolução

- Sistemas de Computação
 - 1^a geração (1945/1955) – Válvulas
 - 2^a geração (1955/1965) – Transistores, *batch*
 - 3^a geração (1965/1980) – ICs, *time-sharing*
 - 4^a geração (1980/ ?) – PCs, workstations, servidores, modelo Cliente/Servidor
- Hoje temos...
 - Datacenters, servidores, portáteis, tablets, smartphones...
 - Tudo interligado em “Cloud”
 - Internet of things (IOT)...

“The network IS the system”



22