

Operating Systems 2021/2022

T Class 01 - Operating Systems Course Overview

Vasco Pereira (vasco@dei.uc.pt)

Dep. Eng. Informática da Faculdade de Ciências e Tecnologia da Universidade de Coimbra

operating system

noun

the collection of software that directs a computer's operations, controlling and scheduling the execution of other programs, and managing storage, input/output, and communication resources.

Abbreviation: OS

Source: Dictionary.com

1 2 9 0



Departamento de
Engenharia Informática
FACULDADE DE
CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE
COIMBRA

Updated on
17 February 2022

Docentes

- Aulas Teórico, Teórico-Práticas e Práticas Laboratoriais
 - Vasco Pereira (vasco@dei.uc.pt), D2.16
 - <https://videoconf-colibri.zoom.us/my/vascopereira>
- Aulas Práticas Laboratoriais
 - David Abreu (dabreu@dei.uc.pt)
 - Frederico Cerveira (fmduarte@dei.uc.pt)
 - José Flora (jeflora@dei.uc.pt)
 - Pedro Neves (pedroneves@dei.uc.pt)

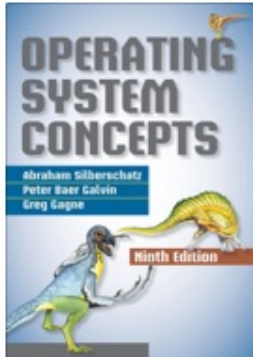
Contacto com docentes

- Horário de gabinete e demais contactos no Nónio;
- Os alunos devem usar os e-mails oficiais da UC, pois só assim se sabe quem é que está a enviar o mail;
 - E-mail oficial da UC ou e-mail da student
 - Endereços do gmail, hotmail, etc! podem não ser respondidos
 - No assunto deve ter o nome da cadeira: “SO” ou “Sistemas Operativos”

Resumo e Objectivos da Cadeira

- O **Sistema Operativo** é um “programa especial” que permite isolar o hardware dos programas que executam no computador:
 - Gestão da memória
 - Gestão dos discos
 - Gestão dos periféricos (teclado, rato, placa gráfica)
 - Gestão dos utilizadores e programas, protegendo todo o sistema
 - ... tudo para que o programador não tenha que o fazer
- **OBJECTIVOS**
 - Saber como funciona um sistema operativo “por dentro”
 - Saber utilizar e programar utilizando as funções fornecidas pelos sistemas operativos, tirando partido das suas potencialidades
 - Aprender programação concorrente
 - Aprender programação de sistema (usando linguagem “C”)

Bibliografia – Parte Teórica



- **Operating System Concepts**

8th Edition [9th Edition (2012), 10th (2018)]

Abraham Silberschatz, Peter Baer Galvin, Greg Gagne

John Wiley & Sons, ISBN 978-1-118-06333-0

[Silberschatz13]

A matéria não se esgota nos slides!!

Bibliografia – Parte Teórica (livro em Português)



- **Sistemas Operativos**

2ª Edição (2012)

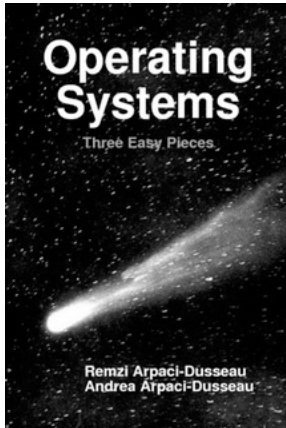
José Alves Marques, Paulo Ferreira, Carlos Ribeiro, Luís Veiga, Rodrigo Rodrigues

ISBN: 978-972-722-756-3

<http://www.fca.pt>

Bibliografia – Parte Teórica

[free online]

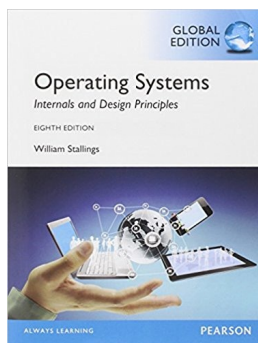


- **Operating Systems: Three Easy Pieces**
Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau
Arpaci-Dusseau Books

<http://www.ostep.org/>
(<http://pages.cs.wisc.edu/~remzi/OSTEP/>)

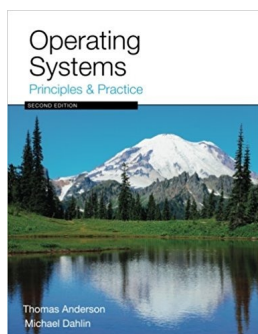
Bibliografia – Parte Teórica

[outros livros de consulta]



- **Operating Systems - Internals and Design Principles**

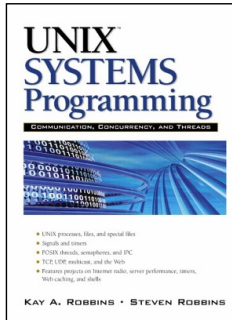
8th Edition (2015) [9th Edition (2017)]
William Stallings
Pearson
[Stallings15]



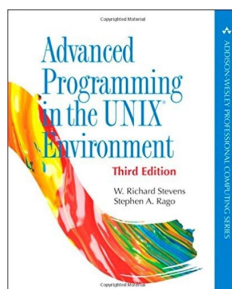
- **Operating Systems - Principles & Practice**

2nd Edition (2014)
Thomas Anderson, Michael Dahlin
Recursive Books
[AndersonDahlin14]

Bibliografia – Parte Prática

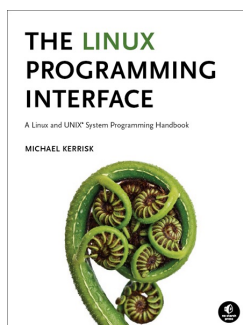


- **Unix Systems Programming: Communication, Concurrency and Threads**
1st Edition(2003), 2nd Edition (2015)
Kay Robbins, Steve Robbins
Prentice Hall
[Robbins03]



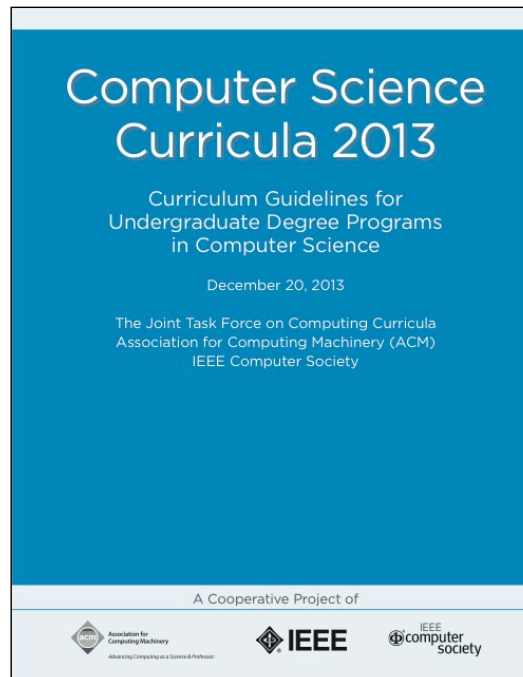
- **Advanced Programming in the UNIX Environment**
2nd, 3rd Edition (2013)
W. Richard Stevens, Stephen A. Rago
Addison-Wesley
[Stevens13]

Bibliografia – Parte Prática (outros livros de consulta)



- **The Linux Programming Interface**
2010
Michael Kerrisk
No Starch Press
[kerrisk10]

Programa recomendado



Aprendizagem fora das aulas

- 2 Documentos de apoio
 - Linux Virtual Machine to support practical exercises
 - C and makefiles
- 3 Tutoriais
 - Installing Linux and the required tools
 - GNU Debugger
 - Shell Scripting (para complementar as aulas TP)

→ Para leitura e execução fora das aulas!

Tópicos abordados nas Aulas Teóricas

1. Funções de um Sistema Operativo
2. Gestão de Processos
3. *Multi-threading*
4. Programação Concorrente
5. Exclusão mútua, sincronização e *deadlocks*
6. Escalonamento de processos
7. Gestão de memória principal/virtual
8. Gestão de armazenamento
9. Sistemas de ficheiros
10. Segurança e Protecção
11. Virtualização

Tópicos abordados nas Aulas Práticas

1. Revisão da programação C em Linux
2. Introdução ao Linux
3. Gestão de processos e threads
4. Prog. Concorrente
5. IPCs: memória partilhada, *pipes*, *named pipes*, filas de mensagens, *memory-mapped files*, semáforos
6. Sinais

Aulas T, TP e PL

■ Aulas T

- Exposição de matéria teórica
- Resolução de exercícios exemplo

■ Aulas TP

- Exposição de matéria prática
- Resolução de exercícios práticos e teóricos

Slides e material
de apoio em
inglês!

■ Aulas PL

- Apoio à resolução de problemas/fichas
- Apoio ao projeto
- Avaliações do projeto
- A matéria das aulas PL acompanha a das aulas TP

Notas adicionais sobre aulas PL

■ Aulas de apoio à resolução de Fichas Práticas

- Pequena introdução da matéria pelos docentes
- Resolução acompanhada do primeiro exercício
- Acompanhamento e ajuda na resolução dos restantes exercícios
 - **O objetivo é serem os alunos a resolver os exercícios!**
- Apenas parte dos exercícios de cada Ficha Prática será resolvido durante a aula; os restantes são trabalho de casa - a sua resolução pode ser discutida com o professor das PL em qualquer aula seguinte

■ Aulas de apoio ao projeto

- Acompanhamento do desenvolvimento do projeto
- Esclarecimento de dúvidas

Fichas Práticas

- 8 Fichas práticas
- Resolução de problemas com base na matéria das TPs
- A realizar nas aulas PL com apoio dos docentes
- Não são avaliadas

- Para aproveitar as aulas PL os alunos devem:
 - Ler a Ficha antes da aula e rever a aula TP em que a matéria foi abordada
 - Ter disponíveis os slides das TPs para consulta

Projecto prático

- **Deverá exercitar os conhecimento teórico-práticos da cadeira e poderá/deverá validar competências na programação de sistemas:**
 - Unix; Processos; Files; Signals; Pipes; Select; Processos; Shared Memory; Semaphores; Memory-Mapped Files; Threads; Sincronização; Condition Variables; Filas de Mensagens
- Grupos de **2 alunos (máximo)** - grupos de 1 apenas mediante justificação
- Submetido via **Inforestudante**
- O projeto deve funcionar na máquina virtual fornecida!
 - Alternativa: funcionar na student2.dei.uc.pt
- Trabalhadores estudantes **devem** entregar trabalhos nos mesmos prazos. As defesas podem ser feitas em horário a acordar com os docentes.
- Todos os trabalhos serão escrutinados para detecção de cópias de código
 - Qualquer cópia de código conduzirá a um processo formal junto da UC!

Componente prática

Pré-Requisitos e Software

- Programação concorrente em “C”, em ambiente Unix/Linux
 - A linguagem C é considerada pré-requisito.
 - O ambiente LINUX/UNIX é fundamental.
 - Deverá ser usada a bibliografia de apoio e tutoriais.
- Será necessário usar uma distribuição Linux
 - Foi escolhido o Ubuntu (ubuntu.com) / Lubuntu (lubuntu.me) para a aulas
 - Pode ser usada uma máquina virtual (e.g. VMWARE, VirtualBox)
 - **Aconselha-se o uso da imagem da máquina virtual fornecida**
- Caso necessitem de uma máquina do DEI:
 - **DEVEM UTILIZAR A...** **STUDENT2.DEI.UC.PT**
 - **NUNCA A ...** **STUDENT.DEI.UC.PT**

Presença nas aulas T, TP e PL

- **Presenças nas aulas**
 - Presenças registadas no UC Student (<https://ucstudent.uc.pt/>)
 - A presença nas aulas é fortemente aconselhada, mas não obrigatória, **excepto** em momentos de **avaliação**

Modelo de Avaliação

■ Avaliação Periódica

- Testes escritos individuais (65% da nota total)
 - 1 Teste teórico (a realizar na Ép. exames normal) (8 valores) (40%)
 - 1 Teste prático (5 valores) (25%)
- Realização de projeto (35% da nota total)
 - 1 Projeto (realizado ao longo do semestre) (7 valores) (35%)

■ Época de Recurso, Especial, Extraordinária

- Pode ser repetido o Teste teórico **e/ou** o Teste prático (Mini-teste + Teste)
- O projecto não é repetível.
 - Se apenas um dos testes for repetido, é usada a nota obtida anteriormente no teste que falta e a nota do Projecto para calcular a nota final para a época de exames em questão.

■ Mínimos

- O **Teste Teórico** e o **Projecto** têm mínimos que correspondem a 35% do peso atribuído a cada uma desses componentes

Modelo de Avaliação

■ Exemplo

- Notas obtidas em Época Normal:
 - 50% Teste Teórico;
 - 60% Teste Prático;
 - 50% Projeto (não repetível nas outras épocas)
 => Nota de Ép. Normal calculada com: 50% Teórico; 60% Prático; 50% projeto
- Notas obtidas em Época Recurso (ex.1)
 - 35% Teste Teórico;
 - 75% Teste Prático
 => Nota de Ép. Recurso calculada com: 35% Teórico; 75% Prático; 50% projecto
- Notas obtidas em Época Recurso (ex.2)
 - Não realiza o Teste Teórico -> será substituído pela nota do teste Teórico de Ép. Normal, caso tenha sido realizado;
 - 75% Teste Prático
 => Nota de Recurso calculada com: 50% Teórico; 75% Prático; 50% projeto
- Mínimos:
 - Caso não tenha mínimos no Projeto não poderá realizar a cadeira em qualquer das épocas (embora possa teoricamente ter acesso a realizar o exame)
 - Caso não tenha mínimos no Teste Teórico não poderá realizar a cadeira nessa época.

Testes escritos

■ Teste Teórico

- Consulta livre em papel
 - não são permitidos quaisquer dispositivos eletrónicos, incluindo telemóveis
- Duração: TBD
- Datas
 - Teste Teórico: ??/??/2022
 - Exame Recurso: ??/??/2022

■ Teste Prático

- Consulta livre em papel
 - não são permitidos quaisquer dispositivos eletrónicos, incluindo telemóveis
- Duração: TBD
- Data
 - 04/05/2022 (a confirmar)

Avaliação do projeto prático

■ Defesa intermédia

- 20% da avaliação final (não recuperáveis na defesa final!)
- Aulas PL
- Semana de 28/03/2022

■ Entrega Final

15/mai/2022

■ Defesa Final (obrigatória)

- 80% da avaliação final
- A partir de 16/mai/2022

Notas sobre Fichas e Projeto

- **Cooperação entre alunos**
 - Os alunos são encorajados a discutir as Fichas e o enunciado do Projeto com os colegas, professores ou quaisquer outras pessoas;
 - Não deve haver partilha do código de soluções;
 - Não podem ser usados repositórios de código que permitam o acesso público!
 - Após perceber o problema em questão, cada aluno (ou grupo) deve construir a sua própria solução;
- Todos os trabalhos submetidos para avaliação serão escrutinados para detecção de cópias.
 - A existência de cópias entre trabalhos levarão a processo disciplinar

Conduta Ética e Fraude

- **Fraude Académica**
- Regulamento 400/2019 - Regulamento Disciplinar dos Estudantes da UC:
 - i) Detectada a situação o docente informa o Diretor, abrindo-se um processo com as provas da eventual fraude.
 - ii) O poder de instauração do procedimento disciplinar pertence ao reitor, sem prejuízo da possibilidade de delegação nos diretores das unidades orgânicas.
 - iii) Provada a fraude, o aluno é reprovado à cadeira.
 - iv) Possíveis sanções disciplinares:
 - Suspensão temporária das atividades escolares;
 - Suspensão da avaliação escolar pelo período de um ano

Alguns indicadores de anos anteriores

	Resumo de 2018/2019	Resumo de 2019/2020	Resumo de 2020/2021
Inscritos	218	215	227
Entregaram Projeto (2ª meta)	157	169	166
Tiveram mínimos	138	149	138
% alunos aprovados, <u>dos que</u> <u>tiveram mínimos no Projecto</u>	82%	80%	91%

APRESENTAÇÃO:
DOCUMENTOS DE APOIO E
TUTORIAIS

Obrigado! Questões?



I keep six honest serving men. They taught me all I knew. Their names are What and Why and When and How and Where and Who.
—Rudyard Kipling