Systems programming

MEEC /MEAer – 2016/2017

Offering

- MEEC (Electrical Engineering)
  - 3rd / 4th year Computer's Major/Minor
- MEAer (Aerospace Engineering)
  - Avionic

Objectives

- Present concepts of systems programming
  - Direct interaction with the OS
- Increase knowledge on the organization and systems interface
- Explore tools and functionality for good SW development practices

Classes

- Theoretical
  - Presentation of course material
  - Resolution of some problems (with programming)
- Laboratories
  - Resolution of programming problems

Classes

- Theoretical
  - Monday 12:30.. 14:00 – EA2
  - Wednesday 11:00 – 12:30 - EA1
- Laboratories
  - SCDEEC
  - Wednesday – 12:30 – 14:30
  - Friday – 9:00 – 11:00
  - Friday – 11:00 – 13:00
  - Registration on FENIX
    - Only for studenst not yet registered

Teaching Staff

- João Nuno Silva
  - joao.n.silva@tecnico.ulisboa.pt
  - 213100347
- Ricardo Martins
  - ricmartins@lx.it.pt

Student profile

- Course
  - MEEC
  - MEAer
  - MEFT
  - MMA
- Previous knowledge
  - Programming language
    - C
  - Operating Systems (user level)
  - Networking sockets (not required)

Evaluation

- Project
  - 50% of final grade
  - Minimum grade – 10
  - Week of 25th of May
    - presentation and discussion of projects
- Exam
  - 50% of final grade
  - Minimum grade - 9

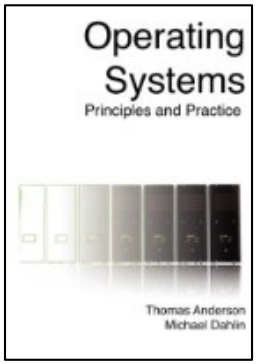
Academic honesty

- In PSIS plagiarism includes:
  - Use of ideas, code or solutions from other students, person or source, without propers credit acknowledgement
- Students should discuss their problems with other colleagues, but should mention that in the submitted work. This discussion will not lower the grade
- BUT:
  - Students should not copy code from or provide solutions to other colleagues

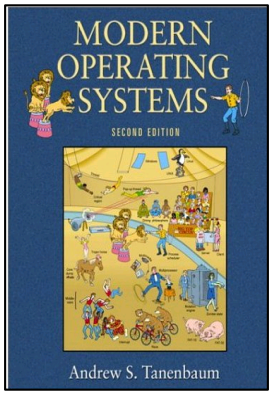
Academic honesty

- Academic dishonesty also included copying in test or exams
- Exams are solved without any external help(communication of printed material)
-
- Academic dishonesty is considered fraud.
- The immediate consequence is the automatic fail in the evaluation where a copy of fraud was detected

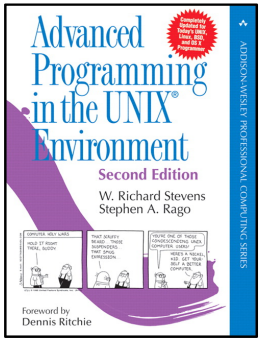
Bibliography



- Operating Systems: Principles and Practice
  - Michael Dahlin, Thomas Anderson
- Modern Operating Systems; A.S. Tanenbaum; Prentice-Hall
  - Theoretical classes



- Advanced Programming in the UNIX Environment ; W. Richard Stevens, Stephen A. Rago; Addison-Wesley
  - Used in laboratory and project



- Other
  - Papers, book chapters
  - Provided in the FENIX

System

- From wikipedia
  - A system is a set of interacting or interdependent components forming an integrated whole.
  - Delineated by its spatial and temporal boundaries,
  - Surrounded and influenced by its environment
  - Described by its structure and purpose
  - Expressed in its functioning.

System Programming

- Understanding of a System
  - Operating System
- Use of a System
  - Operating System
- Development/implementation of a System

Systems programming

- Fundamental in the area of computer networks
  - And Distributed Systems / Internet
- Presents the interface and connection between:
  - Hardware (CPU/ memory/ peripherals)
  - Applications
- Relevant in other areas
  - Telecommunication
  - Control

Program

- Systems
- Architectures and patterns
- Operating System Architecture
- OS programming
  - Nodes
    - Process management
    - Threads
    - Synchronization
  - Communication
    - Shared memory
    - Inter-process communication
    - Data Interoperability
- Performance evaluation
- Event based programming
- Introduction to Software engineering
  - Requirements
  - Software development
  - UML
- Software testing

- C review
- Python + C
- C + assembly
- Process and shared memory
- Threads
- Synchronization
- IPC
- Performance evaluation
- Thrift/ProtocolBuffers/Avro
- SW development tools
- Software testing