Operating Systems

Introduction to Operating Systems

Pietro Laface
Dipartimento di Automatica e Informatica
Politecnico di Torino

Introduction

Operating Systems (01JEZBV)

Pietro Laface
Tel. 011 564 7004
Pietro.Laface@polito.it

Introduction

- Information, programs, tutorials, previous exam texts, labs assignments and solutions, can be found at the address:

http://areeweb.polito.it/didattica/operating_systems/OS/index.html

Easier googling for "Operating Systems Laface"

- Slides will be available on the official site of the course

Introduction

- Organization
  - Schedule
    - Lesson/Practice: 2 blocks of 1.5 hours
    - Laboratory: 1 block of 1.5 hours
  - There is no formal distinction between teaching and practice hours
    - Theory is introduced, and examples and exercises can be illustrated in the same block
  - Laboratory
    - Practice with Linux operating system
    - Application of the theoretical aspects on Linux
    - Script programming (bash and awk)

Introduction

- Topics
  - Introduction to Operating Systems
  - Processes (concept, control, signals, IPC, etc.)
  - Thread (concept, pthread library, etc.)
  - Synchronization (s/w, h/w, semaphores, etc.)
  - Deadlock
  - Linux environment
    - Commands and system administration
    - Shell (UNIX/Linux command interpreter)
    - Scripting languages (bash and awk)

Introduction

- Topics
  - *Preparing for Google Technical Internship Interviews*
    - *...*
    - *Operating systems*
      - *You should understand processes, threads, concurrency issues, locks, mutexes, semaphores, monitors and how they all work. Understand deadlock, livelock and how to avoid them. Know what resources a process needs and a thread needs. Understand how context switching works, how it's initiated by the operating system and underlying hardware. Know a little about scheduling. The world is rapidly moving towards multi-core, so know the fundamentals of "modern" concurrency constructs*

Introduction

- Textbooks
  - Operating System Concepts, Ninth Edition

    A. Silberschatz, P. Baer, and G. Gagne

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

- Other useful books
  - W. R. Stevens: "Advanced programming in the UNIX Environment", Addison-Wesley Publishing Company
  - D. Bovet and M. Cesati: "Understanding the LINUX Kernel- III Edition ", O'REILLY

Introduction

- Exam rules
  - There are no intermediate tests, exemption, or alternative projects
  - The access to the examinations rules are defined and described in the "Manifesto degli Studi" of the current academic year on any subsequent amendments thereof
  - It is essential to book to be admitted at an exam

Introduction

- Examination
  - Written test lasting 100 minutes
  - There is no oral, or possibility of oral
  - During the test is permitted
    - The use of **3 sheets** ("approved forms") provided by the teacher (commands, bash, awk) in original (not handwritten)
    - **No other material is allowed** (no notes, no textbooks, no computers, no cell phones, etc.).
  - The test includes **6 questions**
    - **3 / 4** questions/theoretical exercises
    - **2 / 3** questions/exercises on Linux (commands / scripts)
  - You can withdraw during the test

Introduction

- Evaluation of the examination
  - Each exercise has a weight equal to **6** points
    - 6x6 = 36 ...
  - Both the theory and the Linux parts have a minimum threshold
    - The threshold is about 50% of the cumulated points for each part
    - The exam is passed if **both** the theory and the Linux parts are nor below the threshold
  - The final score is the sum of the points obtained in each question/exercise

Introduction

- Laboratory
  - It is compulsory to deliver a report of the work done every week in the laboratory.
  - The deadline for the delivery of the reports is two weeks from the assignment.
  - if the laboratory reports are missing, incomplete, or not given in time you cannot be admitted to the exam.
  - The reports are part of the exam, but do not contribute to increase the score obtained during the examination
  - On the contrary, your score can be reduced if your reports demonstrate that the laboratory assignments have not been properly done

Example of theory question

- Show **the process generation tree and the output lines** produced by the execution of this program:

```
#include …
int main (){
  int i, j=-1;
  for (i=0;i<3;i++){
    printf ("i=%d\n", i);
    if (!fork()){
      for (j=2;j>0;j--){
        fork();
        execlp ("echo", "i", "j", (char *) 0);
      }
    }
  }
}
```

Example of practice question

- Implement a **bash** script that takes two arguments, **n** and **f**. The script has to append in file **f** the content of the first **n** regular files in the user home directory tree. The script will also append at the end of the content of each file, its basename. The script must produce a warning message on the console if it finds less than **n** files.