Informatique

Operating Systems

IDENTIFICATION

CODE :	IST-4-OPS
ECTS :	3.0

HOURS

Lectures :	0.0 h
Seminars :	20.0 h
Laboratory :	0.0 h
Project :	0.0 h
Teacher-student contact :	20.0 h
Personal work :	20.0 h
Total :	40.0 h

ASSESSMENT METHOD

- mixed written+practical exam (laptops allowed) in the last 2h session

TEACHING AIDS

TEACHING LANGUAGE

English

CONTACT

M. SALAGNAC Guillaume
guillaume.salagnac@insa-lyon.fr
Phone : 0472436413

AIMS

The aim of this course is to acquire a concrete understanding of what it means to "execute" a "program" on a "computer". What is an "instruction" ? What is "a variable" ? What does "the processor" do ? What does "assembly programming" mean ? This course is neither a "computer architecture" course (we will *not* discuss implementation details of the hardware) nor an "operating systems" course (we will not talk about advanced topics such as virtual memory or process context switching).

CONTENT

- Information coding: binary numbers, hexadecimal, ascii
- Binary Arithmetics: addition, subtraction, multiplication, two's complement encoding
- The von Neumann Architecture: CPU+Memory, machine language vs assembly, immediates
- Control Flow: control structures (loops, alternatives), mandatory vs conditional jumps, breakpoints
- Addressing modes and memory instructions: direct vs indirect, indexed addressing
- Memory-mapped input/output: peripheral interface through load/store
- Subroutines: call protocol, return address, parameter passing, calling conventions
- The Execution Stack: push/pop instructions, stack pointer, register spilling
- Recursion: Application Binary Interface, fixed/caller-saved/callee-saved/scratch registers

BIBLIOGRAPHY

- Patterson and Hennessy, Computer Organization and Design: The Hardware/Software Interface

PRE-REQUISITE

- Highschool maths: integer arithmetics, boolean logic
- A laptop with Python installed (python 3.6+)