UNIVERSITÀ DI PISA

LABORATORY II

GIOVANNI MANZINI

Anno accademico	2023/24
CdS	COMPUTER SCIENCE
Codice	732AA
CFU	12

Moduli	Settore/i	Tipo	Ore	Docente/i
LABORATORIO II	INF/01	LEZIONI	96	GIOVANNI MANZINI

Learning outcomes

Knowledge
The student will acquire knowledge related to system programming in C, concurrent programming, use of Makefile's, Python scripting, Assembler programming.

Assessment criteria of knowledge
The knowledge acquired will be assessed using:

- Programming tests in class.
- Programming homeworks.
- Project.
- Final oral exam.

The assessment criteria can be change because of Covid related restrictions.

Skills
The student will acquire skills in:

- Development of C programming language.
- Design and evelopment of multithreading solutions in C.
- Development of Python Scripts.
- Handling of inter process communications even involving different programming languages
- Development of ARM Assembler programs.

Assessment criteria of skills
Programming exercises at home and in class, final project.

Behaviors
At the end of the course students should be able to efficiently tackle complex tasks, possibily using concurrent programs.

Assessment criteria of behaviors
Final project and oral exam

Prerequisites
Good knowledge of at least one imperative language
Knowlegde of elementary data structures: (sorted) arrays and lists, queues, stacks, trees, hash tables
Knowledge of the man Linux/Unix shell commands.

Teaching methods
Delivery: frontal lectures in mixed mode (in person and online)
Learning activities:

- attending lectures
- individual study
- solving of programming exercises individually
- attending office hours with the assistants and professors

Attendance: strongly advised but not mandatory.
Teaching methods:

- lectures with slides
- programming exercises

Syllabus

C programming, including using pointers, strings, standard I/O libraries.
C programming using system calls.
GDB debugging and memroy check using valgrind.
Using Makefile.
Concurrent programming in C using processes and threads.
Communication and sincronisation among processes and threads.
ARM Assembler programming
Python Scripting.

Bibliography

P. Deitel, H. Deitel. C: how to program. Pearson.
Marc J. Rochkind. Advanced UNIX Programming 2nd Edition, Addison-Wesley Professional Computing Series, 2004.

Assessment methods

The exam will be held in two modalities:

1. Intermediate programming tests during the academic year, including homework and in-class tests, final simplified project, oral exam.
2. For the syudents who did not pass the intermediate evaluation: final project and oral exam.

Updated: 12/09/2023 21:27