



Programme of Course "Calcolatori Elettronici"

- Code: I0645
- Type of course unit: Compulsory (Laurea in Ingegneria dell'Informazione curriculum Comune)
- Level of course unit: Undergraduate Degrees
- Semester: 2

Number of ects credits: (Laurea in Ingegneria dell'Informazione) 6 (workload 150 hours)

Teachers: Laura Tarantino

1	Course objectives	The course aims to provide the knowledge necessary for the understanding of modern computer systems, with regard to both methodological and technological aspects . The course deals with issues related to low-level programming (assembly language and machine language) and issues related to the architectural design (CPU, memory, device, and their connections). The topics are treated both from a methodological point of view and with reference to real processors popular on the market, so as to provide an overview of advanced technological solutions and at the same time theoretical skills that remain valid in the long run notwithstanding changes in the systems and their releases.
2	Course content and learning outcomes (dublin descriptors)	<p>Topics of the module include:</p> <ul style="list-style-type: none"> • REPRESENTING INFORMATION IN THE COMPUTER. Signed and unsigned integer numbers, real numbers and floating point representation, characters. Hexadecimal representation. Arithmetic for computers. Addition and subtraction. Logical operations. Designing a simple arithmetic-logic unit. • LANGUAGE OF THE COMPUTER. The MIPS assembly language: operations and operands, conditional branches to manage if statements and loops, supporting leaf and nested procedures, MIPS addressing modes, arrays, pointers. Representing instructions in the computer. Assembler and linker. Example instruction sets of real processors. • THE BASICS OF LOGIC DESIGN. Gates, truth tables and logic equations. Combinational logic, AND-OR and NAND-NAND circuits. Basic circuits: half-adder, full-adder, coder, decoder, multiplexor, demultiplexor, combinational shifter. Constructing a basic arithmetic-logic unit. Memory elements: Flip-flop D, Registers, Register file, SRAM and DRAM. Timing Methodologies. • THE PROCESSOR (I): BASIC DESIGN. Logic design conventions. Building a datapath. A simple implementation scheme. Assessing the performance: defining and measuring performance, system and user CPU time, instruction count, clock cycle time, the classic CPU performance equation. • MEMORY HIERARCHY. The basics of caches: positioning, locating and replacing blocks in direct-mapping, set-associative and full associative schemes. Accessing a cache, handling cache misses, handling writes, Measuring cache performance, reducing the miss penalty using multi-level caches. Designing the memory system to support caches: the interleaving scheme. Virtual memory. The page table and the TLB. Integrating virtual memory, TLB and caches. Example real memory hierarchies. • CONNECTING PROCESSORS, MEMORY AND I/O DEVICES. Basic notions. • MULTICORES, MULTIPROCESSORS, AND CLUSTERS. Basic notions.
3	Course prerequisites	Basic notions on the computer architecture and on programming languages. Experience in C++ programming and in general in the usage of a personal computer. Knowledge of basic data structures (array, stack).
4	Teaching methods and language	<p>Lectures and exercises</p> <p>Language: Italian</p> <p>Reference textbooks</p> <ul style="list-style-type: none"> • Laura Tarantino, <i>Slides dalle lezioni ed eserciziari</i>. • D.A. Patterson, J.L. Hennessy, <i>Struttura e progetto dei calcolatori</i>, Zanichelli.
5	Assessment methods	Written exam, including both exercises and more theoretical questions

