NATIONAL UNIVERSITY OF ENGINEERING
COLLEGE OF SCIENCES
COMPUTER SCIENCE PROGRAM

CC221 – COMPUTER ARCHITECTURE

I. GENERAL INFORMATION

CODE	: CC221 – Computer Architecture
SEMESTER	: 3
CREDITS	: 4
HOURS PER WEEK	: 6 (Theory - Laboratory)
PREREQUISITES	: CC112 Computer Programming
CONDITION	: Mandatory

II. COURSE DESCRIPTION

Allow the student to properly manage the hardware and software of a computer system. Guarantee the good performance and efficiency of the future codification. Demonstrate that the evolution of computer architecture is a consequence of the progress and development of both hardware and software.

III. LEARNING UNITS

1. Digital Logic and Digital Systems

I: Fundamental building blocks (logic gates, flip-flops, counters, registers).

2. Machine-level Data Representation

I: Bits, bytes and words.
II: Representation of numerical data and numerical bases.

3. Assembly-Level Machine Organization

I: Control unit, search instruction (fetch), decoding, and execution.
II: Instruction sets and types (data manipulation, input and output control).
III: Machine and assembler programming language.
IV: Instruction formats.

4. Organization of Memory and Architecture Systems

I: Coding, data compression and data integrity.
II: Memory hierarchy.
III: Cache memories, (address mapping, block size, replacement policies and storage).
IV: Virtual Memory (paging tables, TBL).

5. Communication and Interface

I: Basics of input and output: buffering, handshaking, programmed inputs and outputs, inputs and outputs managed by interruptions.
II: Interruption structures: vectorized and prioritized, recognition of an interruption.
III: External storage, physical organization and drivers.
IV: Buses: bus protocols, arbitrage, direct memory access (DMA).
V: RAID architectures.

6. Functional Organization

I: Control unit: understanding of embedding in hardware vs. micro-programming
II: Introduction to the parallelism at the instruction level (IPL).

7. Multiprocessing and Alternative Architectures

I: Introduction to SIMD, MIMD, VLIW, EPIC.
II: Systolic architecture.
III: Network interconnection (Hypercube, random exchange, mesh, crossbar).
IV: Shared memory systems.

8. Diagnose and safety tools

I: Evaluation of computer systems
II: Diagnosis of processing systems
III: Security and computer viruses

9. Programming platforms and basic nuclei

I: Integrate the hardware and software part through the implementation of a basic core to handle peripherals, main memory, interruptions, etc.

IV. BIBLIOGRAPHY

- Brey, B. B. The Intel Microprocessors: 8086/8088, 80186, 80286, 80386, 80486, Pentium, Pentium Pro, and Pentium II, Pentium III, Pentium 4, 7th edition. Prentice-Hall. 2005.

- Mano, M. M. Computer System Architecture, 3rd edition. Prentice Hall. 1992.

- Norton, P. Peter Norton's Assembly Language Book for the IBM PC. Peter Norton Foundation Series. Brady Publishing. 1988.

- D. Tullsen, S. Eggers and H. Levy. Simultaneous Multithreading: Maximizing On-Chip Parallelism, in the Proceedings of the 22rd Annual International Symposium on Computer Architecture, June 1995.

- S. Adve and K. Gharachorloo. Shared memory consistency models: a tutorial, in Technical Report WRL-TR 95/7, Digital Western Research Laboratory. September, 1995.

- Carpinelli, J.D. Computer Systems Organization & Architecture. Addison Wesley. 2001.

- Hennesy, J.L. et. al. Computer architecture a quantitative approach, 2ed. 1996.

- Joseph D. Dumas II, Joseph D. Dumas. Computer architecture: fundamentals and principles of computer design. Published by CRC Press. 2006.

- John L. Hennessy, David A. Patterson, Andrea C. Arpaci-Dusseau, Andrea C. Arpaci-Dusseau. Computer architecture: a quantitative approach. Published by Morgan Kaufmann. 2007.