The lecture is an elementary introduction to the principles of construction and operation of computers from the point of view programmer. After completion of the course students should understand what effect it has on the computer architecture application performance, which will implement them in professional practice.

Program:

Software architecture [16h]

- Computer organization (system components, evolution, RISC vs. CISC, limitations).
- Representation of data in computer memory (integer, float, string, audio, image).
- Representation of programs in computer memory (MIPS ISA, debugging, disassembling).
- **Structured Programming (conditions, loops, pile, procedures, indicators, arrays, structures, unions).**

- The process of compiling and linking (symbols sections relocation, ABI ELF files, the system memory).
- CPU operating modes (user model and system resources, and privileged instructions).
- Interrupts and exceptions (service, interrupt vector, priorities, NMI, step mode).
- Access to the device input-output (MMIO, polling, interrupts, DMA).

Memory [6h]

- Organization of memory (access time, the protocol DDR).
- **Magnetic disks and solid-state (multi-zone recording, caching, garbage collection blocks).**

- Cache (organization, policy exchange and storage).
- Address Translation (paging, multi-level page table, TLB).

Processor architecture [4h]

- **Primer microarchitecture (combinational circuits and sequential flip-flop, a multiplexer, ALU).**

- Jednocyklowy processor.
- Pipelining (MIPS instruction processing steps, data and control compulsive gambling).

Parallel Processing [4h]

- Superscalar (architecture Out-of-Order, speculation, renaming registers SMT).
- Predicting jumping (BHT, BTAC, predicting local and global).

- SMP systems (atomic instructions, interprocess communications, cache snooping).

- Processing vector (SIMD vector processors, GPGPU).

Literature:

- Computer Organization and Design: The Hardware / Software Interface ; fifth edition; Hennesy, Patterson

- Computer Systems: A Programmer's Perspective ; third edition; Randal E. Bryant, David R. O'Hallaron

Activities:

Most of the classes to the subject will be conducted exercises in the form of declarations. Others are development laboratories (MIPS assembly language programming) and development projects (simple assembler or MIPS processor simulator, the simulator cache predictor and jumping).