| | |
|---|---|
| **Study program:** | **Electrical and computer engineering – Module: Computer engineering, Information technologies, Engineering management,** |
| **Type and level of studies:** | **Undergraduate academic studies (first level of studies)** |
| **Course unit:** | **Computer architecture** |
| **Teacher in charge:** | **Uros Pesovic, teaching assistant Dejan Vujicic** |
| **Language of instruction:** | **English** |
| **ECTS:** | **6** |
| **Prerequisites:** | - |
| **Semester:** | **Summer** |

**Course unit objective**

Familiarizing yourself with the operating mode of a classic von Neumann computer, the types of operations and the types and structure of the data they process; familiarization with the hierarchy of the computer's memory subsystem and management principles; familiarization with the principles of realization of input / output operations and transfer of data within the computer and between the computer and the environment; acquaintance with those aspects of computer architecture that are necessary for acquiring knowledge in other areas of computer engineering such as computer networks and operating systems.

**Learning outcomes of Course unit**

The student can describe the classic von Neumann machine and its basic functional units; explain how the instructions are executed and how they are presented both at the machine level and in the context of assembly languages; explain the different instructional formats; write simple machine programs; identify major memory technologies; describe the principles of memory hierarchy and memory management; describe the role of "cache" and virtual memory; explain how interrupts are used to manage input / output and data transmission; identify different types of buses in a computer system.

**Course unit contents**

*Theoretical classes*

The basic organization of von Neumann's machine.
Control unit; retrieve, decode, and execute instructions.
Sets and types of instruction (data manipulation, control instructions, input / output instructions). Instruction formats. Addressing modes. Input / output operations and interrupts.
Subroutine calls and subroutine return mechanism. Machine programming.
Memory systems and their technologies. Memory hierarchy. Organization of RAM. Cache. Virtual memory.
Programmed input / output. Input / output controlled by interrupts. Highways and Arbitration. Direct memory access.

*Practical classes*

Practical application and validation of acquired knowledge by working with specific computer architectures and their simulators.

**Literature:**

| | |
|---|---|
| 1. | Kip Irvine, *Assembly Language for x86 Processors* |
| 2. | William Stallings, *Computer organization and architecture*, Prentice Hall. |
| 3. | |
| 4. | |
| 5. | |

**Number of active teaching hours**

| Lectures: | Practice: | Other forms of classes: | Other classes | Independent work: |
|---|---|---|---|---|
| 3 | 2 | 0 | | Case study: |

**Teaching methods:**

Realization of lectures on the model of interactive teaching using the methods of practical work.

**Examination methods ( maximum 100 points)**

| **Exam prerequisites** | **No. of points:** | **Final exam** | **No. of points:** |
|---|---|---|---|
| Student's activity during lectures | 5 | oral examination | 20 |
| Practical classes | 15 | written examination | 20 |
| Colloquiums | 30 | | |
| Seminars/homework | 10 | | |

**Grading system**

| **Grade** | **No. of points:** | **Description** |
|---|---|---|
| **10** | **91-100** | Excellent |
| **9** | **81-90** | Exceptionally good |
| **8** | **71-80** | Very good |
| **7** | **61-70** | Good |
| **6** | **51-60** | Passing |
| **5** | **less than 50** | Failing |