Study programme(s): Computer science	

Level: bachelor academic studies

Course title: Operating systems 1

Lecturer: Zoran D. Budimac

Status: obligatory

ECTS: 8

Requirements: completed course "Object-oriented programming"; completed course "Data Structure and Algorithms"

Learning objectives
To introduce students to the basic concepts of operating systems, processes and process management, memory management, external memory management, and file systems.

Learning outcomes
Minimum: Minimum: At the end of the course, it is expected that the successful student is able to understand the concepts, algorithms, structure and principles of modern operating systems.
Desirable: At the end of the course, it is expected that the successful student is able to understand and apply concepts, algorithms, structure and principles of modern operating systems.

Syllabus
Theoretical instruction
The role and development of operating systems. History and types of operating systems. Programming languages suitable for implementation of operating systems. Concurrent programming. Processes and process management. The concept of the process and process states. The process implementation. Interprocess communication and synchronization. Process communication. Process scheduling. Memory management. Managing available internal memory. Organization and management of virtual memory. File Management. The structure of files and directories. File systems.
Practical instruction
Concurrent programming and interprocess communication and synchronization. Simulation of typical algorithms for operating systems.

Literature
Recomended
1. Andrew S. Tanenbaum: Modern Operating Systems, 4th Edition, Prentice Hall, 2015.
2. Deitel, H., Deitel, P., Choffness, D., Operating systems, 3rd edition, Pearson, 2004.
3. Золтан Гелер, Белешке са предавања из предмета Оперативни системи, скрипта,
http://perun.pmf.uns.ac.rs/budimac/os/geller.pdf

Weekly teaching load

Lectures: 3	Exercises: 2	Practical Exercises: 2	Student research: 0	Other: 0

Teaching methodology
During lectures, the classical methodology is applied, through the usage of beam-projector and slides. During theoretical exercises, main principles are outlined and practiced and typical problems and their solutions are analyzed. During practical exercises, students independently apply the mastered techniques.
Knowledge of students is assessed through their ability to apply gained knowledge on appropriate real life problems and is shown during practical exercises. Practical exercises are designed so that two types of classes alternate weekly: classes on which, with the help of assistants, students practice the principles and techniques, discuss solutions and the like, and classes on which students work independently on small projects and their results are studied in detail and then evaluated. On the oral part of the exam students demonstrate a comprehensive understanding of concepts, data structures and algorithms used in operating systems implementation.
The course is supported by many additional resources and specially prepared exercises available in the form of an electronic course on the site of the Department, with a goal of continuous innovation of teaching and encouragement of students to independently research current topics, critically think and to detect application areas of learned materials.

Grading method (maximal number of points 100)

Pre-exam obligations	points	Final exam	points
Practical instruction	50	Oral exam	50