Course Unit Descriptor

Study Programme: Information Technologies

Course Unit Title: System Programming

Course Unit Code: IT302

Name of Lecturer(s): Vladimir Kurbalija

Type and Level of Studies: Bachelor Academic Degree

Course Status (compulsory/elective): Compulsory

Semester (winter/summer): Summer

Language of instruction: Serbian (primary), English (secondary)

Mode of course unit delivery (face-to-face/distance learning): Face-to-face

Number of ECTS Allocated: 4

Prerequisites: None

Course Aims:

The objective of the course is to enable the students to learn and apply programming language C, both in ordinarily and low-level programming.

Learning Outcomes:

Minimum: At the end of the course, it is expected that a successful student is able to realize basic tasks which include dynamic data structure using pointers in programming language C. Furthermore, the usage of UNIX system calls is expected.

Desirable: At the end of the course, it is expected that a successful student is able to realize advanced tasks which include advanced dynamic data structure using pointers in programming language C. Furthermore, a deep understanding of memory management as well as the usage of UNIX system calls in an advanced manner is expected.

Syllabus:

Theory

Introduction to programming language C. Control flow statements. Arrays and strings. Functions and parameter passing methods. Pointers, memory allocation and deallocation. Structures, unions and bit fields. Files and file oriented functions. Dynamic data structure: list, tree, stack. UNIX system calls.

Practice

Implementation of various problems in C. Implementation of various data structures (list, tree, stack). UNIX system calls.

Required Reading:

1. Милан Чабаркапа, Ц – основи програмирања, Круг, Београд, 1996.
2. Andrew S. Tanenbaum, Modern Operating Systems, 2nd Edition, Prentice Hall, 2001.
3. Adam Hoover, System Programming with C and Unix, 1st Edition, Pearson, 2009.

Weekly Contact Hours: 3	**Lectures:** 1	**Practical work:** 2

Teaching Methods:

During lectures, the classical methodology is applied, through the usage of beam-projector and slides. 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. On the oral part of the exam students demonstrate a comprehensive understanding of concepts, data structures and algorithms which are presented.

Knowledge Assessment (maximum of 100 points): 100			
Pre-exam obligations	points	**Final exam**	points
Active class participation		written exam	
Practical work	60	oral exam	40
Preliminary exam(s)		…….	
Seminar(s)			
The methods of knowledge assessment may differ; the table presents only some of the options: written exam, oral exam, project presentation, seminars, etc.			