Course Unit Descriptor

Study Programme: Information Technologies

Course Unit Title: Programming Paradigms

Course Unit Code: IT604

Name of Lecturer(s): Mirjana Ivanović

Type and Level of Studies: Bachelor Academic Degree

Course Status (compulsory/elective): Elective

Semester (winter/summer): Winter

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

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

Number of ECTS Allocated: 7

Prerequisites: None

Course Aims:

Presentation to students historical and practical reasons for development of a lot of different programming languages. Clarification of specific characteristics, similarities and differences between several widely accepted programming paradigms (object oriented, functional, logical multi-paradigm). Detailed description of several key representative languages from each of mentioned paradigms.

Learning Outcomes:

Minimal: A the end of the course, it is expected from a successful student to demonstrate general understanding of concepts of several programming languages and understand significance of different programming styles.

Optimal: At the end of the course, it is expected from a successful student to be able to understand concepts of programming languages, understand significance of different programming styles and demonstarte skills of developing and coding programs in several programming paradigms.

Syllabus:

Theory

Hystory of programming languages developement. Procedural and non-procedural programming languages. Characteristics and usual differences between programming languages. Detailed description and comparison of several programming styles (functional, logical, multiparadigms) and key characteristics of programming language representatives. Sintax and semantics. Basic notions and mathematical background. Data structures.

Practice

Comparative approach in implementation of typical programming tasks and algorithms. Illustration of philosophies that are in essence of programming paradigms and languages presented at theoretical classes. Testing of already existing solutions in different languages, tools, possible applications et cetera. Different programming tasks for each student connected to data types, statements, data structures and so on.

Required Reading:

1. Budimac Z., Ivanović M., Bađonski M., Tošić D: "Programming language Scheme," University of Novi Sad, Faculty of Sciences, Novi Sad, 1998.
2. Radovan, M: "Programming in PROLOG", Informator, Zagreb, 1987.
3. Tošić D., Protić R., "PROLOG through examples", Tehnical book, Belgrade, 1991.
4. Prince P., Craford T., "C for programmers", Micro book, Belgrade, 2006.
5. Odersky M., Spoon L., and Venners B., "Programming in Scala", Addison-Wesley, 2016.

Weekly Contact Hours: 5	**Lectures:** 2	**Practical work:** 3

Teaching Methods:

At lectures, classical teaching methodology is applied, with usage of a beam-projector and slides. Essential principles og programming paradigms are illustrated using characteristic examples. During theoretical excersises principles presented on theoretical classes are illustrated, typical problems and their solutions are analysed. After that students can model their own solutions. Acquired knowledge and skills are evaluated via 3 tests while during practical classes students work individually on small projects. During oral exam students have to show understanding of several programming styles.

Knowledge Assessment (maximum of 100 points):

Pre-exam obligations	points	**Final exam**	points
Tests	30	written exam	
Practical tasks	30	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.