SYLLABUS

1. Course title:

Compiler Design

2. Code:

RI402

3. Cycle of study:

1

4. ECTS credits:

6

5. Type of course:

● Mandatory ○ Elective

6. Prerequisites:

[RI201] Computer Architecture, [RI301] Data Structures

7. Class restrictions:

8. Duration / semester:

1 7

9. Weekly contact hours:

9.1. Lectures: 3

9.2. Seminars: 1

9.3. Laboratory/Practice classes: 1

10. Faculty:

Faculty of Electrical Engineering

11. Department/study program:

Electrical Engineering and Computer Science

12. Lecturer:

Ph.D. Edin Pjanić, assistant prof.

13. Lecturer's e-mail:

edin.pjanic@untz.ba

14. Web site:

15. Course aims:

The course objective is to teach students theory and practice of design and development of a compiler and interpreter in order to be able to develop a simple programming language.

16. Learning outcomes:

Upon completion of the course students should:
- understand the theory and practice of compilation, in particular, the lexical analysis, parsing and code generation phases of compilation
- analyze a simple programming or other computer language
- design a compiler/interpreter for a simple programming language
- use the learnt techniques when faces with real problems that demand parsing or similar tasks in other areas of computer and software engineering .

17. Course content:

Programming language basics. Structure of a compiler. Lexical Analysis. Regular expressions. Finite state machines. Parsing. Syntax analysis. Lexer and parser generators. Syntax trees. Control flow analysis. Data flow analysis. Runtime organization. Code generation.

18. Learning methods:

Lectures are accompanied by presentations and other materials which are uploaded on the course web site. Discussions with teaching assistant, laboratory excersizes and homeworks.

19. Assessment methods:

Continuous assessments are performed during the semester (homeworks, quizzes, a project), one test and a final exam. The final exam includes the questions related to the entire content of the course, focusing on the areas that are not covered by the midterm tests.

20. Assessment components:

Grading scale:
Homeworks, quizes: 50%
Midterm tests: 30%
Final exam: 20%
The final grade is based on the continuous assessments that are performed during the semester (homeworks, quizzes) and the final exam.

21. Required reading list:

A. Aho, R. Sethi, J. Ullman, "Compilers: principles, techniques, and tools", 2nd ed., Pearson Education, 2006

22. Web sources:

23. Applicable starting from the academic year:

2016/2017

24. Adopted in the Faculty/Academy session:

04.04.2016