Automata theory and compilers

Academic Year: (2018 / 2019) **Review date: 11-05-2018**

Department assigned to the subject: Computer Science and Engineering Department

Coordinating teacher: SANCHIS DE MIGUEL, MARIA ARACELI

Type: Compulsory ECTS Credits : 6.0

Year : 2 Semester : 1

REQUIREMENTS (SUBJECTS THAT ARE ASSUMED TO BE KNOWN)

Programming
Algorithms and Data Structures

OBJECTIVES

The aim of this course is that students acquire the following skills:
EURO-ACE SKILLS
1. Basic competences

CB5: Learning abilities required to undertake later studies with a high degree of autonomy.

2. Transverse and generic competences

CGB3. Ability to understand and control basic concepts of Discrete mathematics, Logics, Algorithms and Computational Complexity, and their application for the resolution of problems related to Engineering.

3. Common competences to Computer Science.

CECRI6. Knowledge and application of basic algorithms and procedures of Computer Science to design solutions to problems, and analyze the suitability and complexity of the proposed algorithms.

CECRI15. Knowledge and application of basic techniques and principles of intelligent systems and their practical application.

DESCRIPTION OF CONTENTS: PROGRAMME

1.Introduction to the theory of automata and formal languages.
- 1.1. Why study Automata Theory. History and Origins
- 1.2. Relationship with others Areas of Knowledge
- 1.3. Machines, Languages and Algorithms.

2.- Automata Theory
- 2.1 Introduction and Definitions.
- 2.2 Mathematical model of an automaton
- 2.3 Automata and algorithms.
- 2.4 Discrete, continuous, and hybrid automata. Classes of automata.

3.Finite Automata
- 3.1 Definition and Representation of Deterministic Finite Automata (DFA)
- 3.2. DFA as Recognition Device
- 3.3. Equivalence and Minimization of DFA
- 3.4. Theorems of DFA
- 3.5. Definition and Representation of Nondeterministic Finite Automata (NDFA)
- 3.6. The Language of a NDFA
- 3.7. Equivalence of DFA and NDFA

4.Languages and Formal Grammars.
- 4.1. Operations with Words. Operations with Languages. Derivations.
- 4.2 Concept of Grammar. Formal Grammar.
- 4.3. Chomsky Hierarchy and Equivalent Grammar

4.4 Context-Free Grammar
4.5. Language of a Context-Free Grammar. Parse Tree
4.6. Well-Formed Grammar
4.7. Chomsky Normal Form. Greibach Normal Form

5.Regular Languages.
5.1. Definition of Regular Languages
5.2. DFA for a Regular Grammar
5.3. Equivalence of Regular Expressions
5.4. Kleene's Theorem
5.5. Characteristic equations
5.6. Synthesis Problem: Recursive Algorithm
5.7. Derivatives of Regular Expressions

6.Pushdown Automata.
6.1. Definition of Pushdown Automata (PDA).
6.2. Transitions, Movement and Instantaneous Description in PDA.
6.3. Acceptance by Empty Stack. Acceptance by Final State.
6.4. Language Accepted by a PDA.
Equivalence of PDA by Empty Stack and PDA by Final State.
6.5. From Context-Free Grammar to Push-Down Automata.
6.6. From Pushdown Automata to Context-Free Grammar.

7.Turing Machine.
7.1. Definition if Turing Machine.
7.2. Variations of Turing Machine.
7.3. Universal Turing Machine.

8.Computational Complexity
8.1.Complexity Theory
8.2.Complexity of algorithms
8.3.P versus NP problems
8.4 Defining complexity classes
8.5 Time complexity
8.6 Hierarchy theorems
8.7 Non-computational problems
8.9 Limits of Computability

LEARNING ACTIVITIES AND METHODOLOGY

Theoretical lectures: 1.5 ECTS.
PO: a,c,e,g
These clases constitute a guide for students to achieve cognitive skills, and acquire the basic elements to develop procedural skills. A part of the ECTS corresponds to the load of autonomous work carried out by the students.

Exercices and practical classes (Exercises, Problems and Practices): 2 ECTS.

ASSESSMENT SYSTEM

The evaluation will consist in continuous assessment and a final exam.

The aim of continuous assessment is to help the students keep track of their learning progress, obtaining continous feedback about the competences acquired during the semester. This way,mid-term exams and practices are intended to be both learning and evaluation activities.

The aim of the final exam is to assess the extent to which the students have acquired the cognitive and procedural competences.

The continuous assessment will sum up 50% of the final grade. The continuous assessment will consist of:
- Three mid-term exams (test questions, short questions and problems),and
- Four practical assignments using JFLAP software tool (http://www.cs.duke.edu/csed/jflap/).

Each mid-term exam will sum up 12% of the final grade
and the practical assessments 14% of the final grade.
PO: a,c,e,g,h,k

The final exam (50 % of grade) will consist in theoretical questions as well as practical exercises.
PO: a,c,e,g

More than 4 points in the final exam will be required to sum continuous assessment.

% end-of-term-examination:	60
% of continuous assessment (assigments, laboratory, practicals…):	40

BASIC BIBLIOGRAPHY

- Enrique Alfonseca Cubero, Manuel Alfonseca Cubero, Roberto Moriyón Salomón. Teoría de autómatas y lenguajes formales., McGraw-Hill, 2007
- John E. Hopcroft, Rajeev Motwani, Jeffrey D.Ullman. Introduction to Automata Theory, Languages, and Computation (Third Edition), Pearson Education, Pearson Addison Wesley.
- Manuel Alfonseca, Justo Sancho, Miguel Martínez Orga. Teoría de lenguajes, gramáticas y autómatas., Publicaciones R.A.E.C. ISBN: 8460560929. , 1997.
- Pedro Isasi, Paloma Martínez y Daniel Borrajo. Lenguajes, Gramáticas y Autómatas. Un enfoque práctico., Addison-Wesley, 1997
- Susan H. Rodger and Thomas W. Finley. JFLAP: An Interactive Formal Languages and Automata Package. , web, 2006