UNIVERSITÀ
DEGLI STUDI DI MILANO-BICOCCA

COURSE SYLLABUS

Languages and Computability

2627-2-E3102Q112

Aims

Knowledge and Understanding
Students will learn the fundamentals of formal language theory and its connections to lexical and syntactic analysis, including those relevant to programming languages.

Applied Knowledge and Understanding
Students will be able to define regular and context-free grammars necessary for the use of lexical and syntactic analyzers. They will also acquire the ability to use analyzers in practical contexts.

Independent Judgment
Students will develop the ability to evaluate which grammar is most appropriate for describing different languages.

Communication Skills
The great attention paid to formal aspects will allow students to understand the importance of unambiguous communication, using the correct terminology to express the notions and concepts learned.

Learning Skills
The formalization of concepts will facilitate deductive learning mechanisms. Furthermore, presenting examples and exercises on the board, immediately after explaining a technique or algorithm, helps clarify any doubts and particular cases.

Contents

Finite automata, regular languages and regular expressions. Context-free languages, context-free grammars and pushdown automata. Elements of the theory of computation: Turing machine, the Church-Turing thesis, the Universal Turing machine, unsolvable problems. Lexical analyzers and parsers.

Detailed program

1. Introduction and motivations. Basic mathematical concepts for automata theory
2. Deterministic finite state automata. Non-deterministic finite state automata. An application: searching in texts. Finite state automata with epsilon-moves
3. Regular expressions. Finite state automata and regular expressions. Applications of regular expressions. Algebraic properties of regular expressions
4. Properties of regular languages. The Pumping Lemma as a tool to (dis)prove regularity of a language. Regular languages closure in respect to boolean operations. Equivalence and minimization of automata
5. Grammars. Context free grammars. Parse trees. Applications of context free grammars. Ambiguity of grammars and of languages. Pumping Lemma for context free grammars
6. Turing Machines. Uncomputable problems. The basic Turing machine. Extensions of the basic Turing machine. Reduced Turing Machines
7. Computability. Non Recursively Enumerable languages. Recursively Enumerable and Recursive languages. Undecidable problems and Turing Machines
8. Lexical and syntactic parsers. Parsing algorithms

Prerequisites

The contents of the first year's courses

Teaching form

28 lessons of 2 hours held in the classroom, in delivery mode, in person.
12 hours held in the laboratory, in delivery mode in the initial part which is aimed at involving students in an interactive way in the subsequent part. All activities are held in person.

The course is delivered in Italian.

Some self-assessment exercises will be published on the eLearning (Moodle) web page.

Textbook and teaching resource

Textbook (the english version is also available):

- J.E. Hopcroft, R. Motwani, J.D. Ullman, Automi, linguaggi e calcolabilità, Addison Wesley
- Dick Grune e Ceriel JH Jacobs. Parsing techniques. Monographs in Computer Science. Springer

Learning material provided on the e-learning platform

Semester

First semester

Assessment method

The learning assessment includes a written exam and an oral exam.
The grade is overall and is determined at the end of the oral exam.

The **written exam** requires completing six exercises similar to those covered in class and available on the course e-learning site. The objective of the written exam is to intensively test students' preparation on key topics from the course program and to assess their problem-solving skills.

Students who pass the written exam are admitted to the oral exam.

The **oral exam** includes a discussion of the written exam and questions on course topics. The objective of the oral exam is to assess the student's ability to present the course topics using the appropriate level of formalism and to make brief, yet correct, arguments about them.

The oral exam covers all course content, including the topics covered in the lab.

The oral exam must be taken **in the same exam session in which the written exam was passed**. Students who have not passed the written exam cannot take the oral exam.

Anyone who fails the oral exam must retake the written exam.

The exam sessions are in November, January, February, June, July, and September. Second-year students can take the full exam starting in January.

During the course, there are two ongoing written tests, held in November and January. These tests have the same format and objectives as the written exam, and cover the first and second halves of the course syllabus, respectively.
Passing each ongoing test allows students to skip the corresponding written exam exercises.
In the February exam session, students can take either the full written exam or one of the two partial exams by completing the three corresponding exercises. Students can retake one of the two partial exams even if they have previously passed it. In this case, the new grade cancels the previous grade.
Similarly, students can take the full written exam even if they have previously passed one of the two partial exams, canceling the previous grade.

Those who pass the two ongoing tests in November and January can choose to take the oral exam in the January or February session.

Office hours

By appointment

Sustainable Development Goals

QUALITY EDUCATION