Universidad Zaragoza

Academic Year/course: 2022/23

30214 - Computer Science Theory

Syllabus Information

Academic Year: 2022/23
Subject: 30214 - Computer Science Theory
Faculty / School: 110 - Escuela de Ingeniería y Arquitectura
326 - Escuela Universitaria Politécnica de Teruel
Degree: 439 - Bachelor's Degree in Informatics Engineering
443 - Bachelor's Degree in Informatics Engineering
ECTS: 6.0
Year: 2
Semester: First semester
Subject Type: Basic Education
Module:

1. General information

1.1. Aims of the course

The course and its expected results respond to the following approaches and objectives:

- Train the student so that he/she can abstract problems to solve using a computer.
- Know the basic computation models on which current computers are based and identify the most adapted to each problem.
- Assimilate paradigms of well-studied problems in the context of computing so that you can reduce or adapt them to the problems that arise.
- Know the capabilities and limitations of automatic problem solving and evaluate the necessary resources for it.

These approaches and objectives are aligned with some of the Sustainable Development Goals, SDGs, of the 2030 Agenda (https://www.un.org/sustainabledevelopment/en/) and certain specific goals, in such a way that the acquisition of the Learning outcomes of the subject provide training and competence to the student to contribute to a certain extent to their achievement:

Goal 1: End poverty in all its forms everywhere.
Target 1.4: By 2030, ensure that all men and women, in particular the poor and the vulnerable, have equal rights to economic resources, as well as access to basic services, ownership and control over land and other forms of property, inheritance, natural resources, appropriate new technology and financial services, including microfinance.

Goal 8: Promote inclusive and sustainable economic growth, employment and decent work for all.
Target 8.2: Achieve higher levels of economic productivity through diversification, technological upgrading and innovation, including through a focus on high-value added and labour-intensive sectors.

Goal 16: Promote just, peaceful and inclusive societies.
Target 16.5: Substantially reduce corruption and bribery in all their forms.

1.2. Context and importance of this course in the degree

The Theory of Computation is a basic course taught in the second year of the degree. This particular temporal location allows students to apply the knowledge acquired in this subject to all courses of the degree: decidability, language theory and complexity. These tools will be part of the set of fundamental skills and methods that the computer science engineer will apply in his work.

1.3. Recommendations to take this course

It is convenient that the student has taken the courses of Programming I (1st Semester) and Discrete Mathematics (2nd Semester).

2. Learning goals

2.1. Competences

After passing the course, the student will be more competent to ...

- Solve problems and make decisions with initiative, creativity, and critical thinking.
- Communicate and transmit knowledge, abilities and skills in Spanish.
- Use the techniques, skills and tools of Engineering necessary for the practice of it.
- Learn continuously and develop autonomous learning strategies.
- Apply information and communication technologies in Engineering.
- Understand and master the basic concepts of discrete mathematics, logic, algorithms and computational complexity, and their application to solve engineering problems.
- Use computers, operating systems, databases and computer programs with applications in engineering.

2.2. Learning goals

The student, to pass this subject, must demonstrate the following results ...

- Know the basic computation models.
- Find the simplest calculation model for each problem.
- Discard wrong solutions as too simple for given problems.
- Adequately describe the computation processes.
- Apply the formalisms of language theory in problem solving.
- Transform informal statements into formal statements and vice versa.
- Know the limitations of automatic problem solving.
- Identify basic unsolvable problems such as the halting problem or the virus detection problem.
- Analyze the cost in time and memory of an algorithm.
- Identify problems that require too many computational resources.

2.3. Importance of learning goals

The set of learning outcomes can be summarized by saying that the student will be able to abstract a problem to be solved using a computer, identifying the most appropriate computation model, reducing it to known problems and identifying the limitations of its resolution and the resources necessary for it. Having learned to do it well and efficiently is of vital importance in the context of Computer Science studies of which one of its pillars is solving computation problems.

3. Assessment (1st and 2nd call)

3.1. Assessment tasks (description of tasks, marking system and assessment criteria)

The student must demonstrate that he has achieved the expected learning outcomes through the following assessment activities

At the Zaragoza School of Engineering and Architecture:

Laboratory practical work (30%). On the one hand, a previous report will be assessed where the student must have identified the information needs to solve the problems raised and their use in the resolution. The critical capacity when selecting alternatives and the degree of justification of the solution reached will also be assessed.

On the other hand, the fluency in the use of the computer to solve problems will be assessed. The solutions implemented for each of the exercises proposed for the practical sessions will also be evaluated, taking into account the quality of the procedures and efficient resolution strategies in the computer.

Written exam (70%) in which questions and/or problems in the field of Computer Science will be proposed to be solved by means of a computer, with a typology and level of complexity similar to that used during the course. The quality and clarity of the resolution strategy, as well as its efficiency, will be assessed.

A minimum mark of 4 points out of a total of 10 is required in the written exam to pass the course. If this minimum grade is obtained, then the written exam weighs 70% in the grade of the course and, if this minimum is not reached, then the grade in the course is that of the written exam.

At the Polytechnic University School of Teruel:

The final grade for the call in the ordinary call is divided as follows:

Written exam. 70% of the final grade. It will consist of a part of theory and another of problems. In the middle of the course there will be a partial test that will allow to "advance" grade for the exam.

Theoretical work. 5% of the final grade. It will consist of a thematic work to be defined during the course that will deal with some of the subjects of the course.

Practical work. 25% of the final grade. There will be several deliverable practical works throughout the course.

A minimum of 4 points out of a total of 10 is required in the written exam to pass the course. If this minimum grade is obtained, then the weighting indicated above will be carried out. If this minimum is not reached, then the grade in the course

is that of the written exam.

In the case of the extraordinary call, the final grade will be the grade of the extraordinary exam, taking into account that this exam will have a part of theory and problems that will be 75% of the final grade, and a part of practical works that it will be 25% of the final grade. The marks of the partial exam and of the theoretical work will not be kept for the extraordinary call.

Organization of evaluation activities

The student will pass the course by carrying out the activities listed in the previous section and with the relative weightings indicated there. The global assessment will be divided into two parts corresponding to the cited activities and the date of its completion will be specified in advance by the center. Students who have passed activity 1) during the course will also be able to present themselves to improve their grade on the dates of the global assessment.

4. Methodology, learning tasks, syllabus and resources

4.1. Methodological overview

The learning process that is designed for this course is based on:

- The presentation of the contents of the subject in lectures in the classroom
- Solving problems in class.
- Personal study of the subject by students.
- The development of lab assignments by students, guided by teachers in the laboratory
- Solving simple problems of increasing difficulty proposed by the teachers.

Keep in mind that the subject has both theoretical and practical orientation. Therefore, the learning process emphasizes both student attendance at lectures, as in the practical work in the laboratory, in solving simple problems of increasing difficulty, and individualized study.

4.2. Learning tasks

The course includes the following learning tasks:

- In classes taught in the classroom, the program of the course will be presented.
- In classes of case studies, problems will be solved using the concepts and techniques presented in the course syllabus.
- Practical work sessions will be held in a computer lab.

4.3. Syllabus

The course will address the following topics. The course topics are organized around three pillars:

(1) Theory of formal languages, with an emphasis on regular languages and context-free languages;

(2) Fundamentals of Computability, to narrow what problems can be solved algorithmically;

(3) Fundamentals of Algorithmic Complexity, to define what is the efficiency of an algorithmic solution and the number of resources an algorithm needs.

- **Topic 0: Preliminaries**
  Mathematical Preliminaries: sets, functions, relations, induction.
  Definition of alphabet and language.

- **Topic 1: Regular Languages**
  Regular expressions and regular languages.
  Deterministic finite automata (DFA) and nondeterministic finite automata (nDFA)
  Equivalence between DFA and nDFA.
  Properties of regular languages. Pumping Lemma.

- **Topic 2: Context-Free Languages**
  Grammars and context-free languages.
  Pushdown automata.
  Simplification of grammars.
  Properties of context-free languages. Pumping Lemma.

- **Topic 3: Computability**
  Turing machines.
  Languages and Turing machines. Church-Turing thesis.

Decidability. Non-decidable problems.

- **Topic 4: Complexity**
  The classes of languages P and EXP.
  The classes of languages NP and NP-complete.

The concepts, methods and tools of the above paragraphs are illustrated through examples, as realistic as possible, within the areas of computer security, cryptography, natural language processing and compression of information.

4.4. Course planning and calendar

Scheduling of sessions and presentation of works

The schedule of the course will be defined by the School in the academic calendar of the corresponding academic year.

Student Work

The dedication of the student to achieve the learning outcomes in this course is estimated at 150 hours distributed as follows:

In the School of Engineering and Architecture of Zaragoza:

- 56 hours, approximately, of classroom activities (lectures, problems and laboratory)
- 40 hours of teamwork
- 51 hours of personal study (study booktexts and lecture notes, problem solving, preparing lessons and lab assignments)
- 3 hours deoted to the written final exam

At the Polytechnic University School of Teruel:

60 hours of classroom activities (lectures, problems and laboratory)
24 hours of teamwork
60 hours of personal study (study booktexts and lecture notes, problem solving, preparing lessons and lab assignments)
6 hours of evaluation activities

4.5. Bibliography and recommended resources

Teruel:

http://psfunizar10.unizar.es/br13/egAsignaturas.php?codigo=30214&Identificador=12956

Zaragoza:

http://psfunizar10.unizar.es/br13/egAsignaturas.php?codigo=30214&Identificador=13381