

## 30213 - Data Structures and Algorithms

### Syllabus Information

**Academic year:** 2023/24

**Subject:** 30213 - Data Structures and Algorithms

**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:** Compulsory

**Module:**

### 1. General information

In this subject the student will improve his ability to design and develop computer programs with emphasis on the identification, design and definition of Abstract Data Types (ADTs) regardless of their implementation. The student will learn how to design and implement TADs to be reusable, efficient and robust, and to implement them guaranteeing these properties. Some of the most frequently used fundamental TADs will be presented, for which different implementation alternatives will be studied and compared.

These approaches and goals are aligned with the following Sustainable Development Goals (SDGs) of the United Nations Agenda 2030 (<https://www.un.org/sustainabledevelopment/es/>), so that the acquisition of the learning results of the subject provides training and competence to contribute in some measure to the achievement of Targets 8.4 and 16.5

To take this subject it is necessary to have previous training in programming, at the corresponding level

to the one required to pass the subject Programming II. It is also very convenient to have a mathematical background at the level of Discrete Mathematics.

### 2. Learning results

R6. Define abstract data types (TAD's) independently of their implementation.

R7. Implement TAD's in a modular programming language.

R8. Implement and use some fundamental TADs, such as stacks, queues, lists, search trees, hash tables and graphs.

R9. Compare different TAD implementation alternatives with respect to algorithm execution time and memory usage.

R10. Tackle the modular design of medium-sized programs by identifying, defining and implementing the necessary TADs.

R11. Apply basic algorithmic schemes (such as divide to win, search with backtracking, voracity...) to problem solving.

### 3. Syllabus

1. Programming with Abstract Data Types.

2. Linear data types.

3. Arborescent data types.

4. Functional data types.

5. Introduction to networks.

6. Introduction to algorithmic schemes.

### 4. Academic activities

**Lectures:** the syllabus will be explained, 30 hours (all hours indicated are approximate)

**Problem classes:** sessions to solve exercises and design and implementation cases, 15 hours

**Personal study of the subject, preparation of classes and practical activities:** continuous study and work from the first day of class, 60 hours first day of class, 60 hours

**Laboratory practices:** 15 hours in the laboratory, in addition to the teamwork required to complete them

**Team work:** to solve the tasks proposed in the practices and whose result is the delivery of the resulting programs conveniently designed, implemented and documented (25 hours each student).

**Evaluation tests:** 5 hours

## 5. Assessment system

The subject will be evaluated in the **global assessment** modality divided in who parts in each call:

- P1, written exam, a minimum grade of 5.0 will be required to pass and mediate with the rest of the parts. The should solve programming and design problems, and answer conceptual questions if necessary. It will be graded with a score from 0 to 10, for which in general the following will be valued: the quality and clarity of the answers and solutions proposed, their adequacy to the specifications and restrictions, the quality of the design, the adequate application of the resolution methods, and the time spent.

- P2, practical part, a minimum grade of 5.5 will be required to pass and mediate with the rest of the parts. There will be two alternative to obtain the grade for this part, referred to here as P2A or P2B.

- P2A: Carrying out team practices during the four-month period of the subject.

Teams will be formed, according to the limits in size and configuration indicated by the professors, which will have to hand in their results for all the programming problems indicated in the practice scripts of the course, complying with the deadlines and indications given for each one of them.

For each student, the final grade obtained with his practical team work P2A, will be used as his grade for the practical part P2, unless he takes an individual test P2B as described below.

- P2B: Individual laboratory practical exam, at each session.

In the practical exam, the student will be presented with programming exercises of a similar nature to those performed in the practicals or those seen in class.

For those students who take the individual exam P2B, their grade obtained in that exam will be used as their grade for the practical part of the subject, P2.

In general, each of the practical deliveries as well as the individual practical exam will be graded with a grade from 0 to 10, for which the following will be evaluated: the correct operation and performance of the programs according to specifications, the quality of their design, the adequate application of the resolution methods, the authorship and originality, and the time spent, as well as the ability of the team members to explain and justify the design.

In order to pass the subject it will be necessary to obtain the minimum grades required, both in P1 and P2, and in such case, for the calculation of the final grade in the subject, P1 will be weighted at 70%, and P2 at 30%.

If the final grades obtained by the student in parts P1 and P2 are both greater than or equal to the minimum grades required in the respective parts, then the student's final grade will be obtained as the weighted sum of their P1 and P2 grades. Otherwise, the student's grade will be equal to the minimum between: a 4.0 or the weighted sum of their P1 and P2 grades.