Subject:	Data Structure and Algorithms	**Code**	72.34
Credits:	6	**Total hours:** 102	
Department	Digital Systems and Data	**Year:**	2020

Course: Computer Science Engineering

Curriculum: S10 A - Rev18, S10-Rev23, S10 - Rev18

Subject presentation:

This subject corresponds to the basic cycle of the Computer Science Engineering degree. Organizations need to manage their information and access it quickly and efficiently.

The aim of this subject is to provide knowledge about different problems found in a digital world and how they can be solved with properly selected algorithms and data structures. It also focuses on the analysis of different ways of solving problems and the selection of the most appropriate one based on the proposed objective. Some of the data structures studied in the course will be developed further, for example in database courses.

Learning objectives:

1) Analyze the potential of different data structures to represent information.
2) Choose the appropriate representation model and algorithm for solving a problem.
3) Characterize heuristics for the approximate solution of a complex problem.

Contents:

No.	Description
1	**Algorithms.** Techniques for the design and analysis of algorithms. Calculation of time and space complexity. Efficiency analysis.
2	**Algorithms for text search** Exact (Knuth-Morris-Pratt) and approximate search methods (phonetic algorithms: soundex, metaphone and their variants; editing: Levenshtein and its variants; tokenisation: Q-grams. Algorithmic strategies for their implementations (brute force, divide & conquer, greedy, dynamic programming). Applications: creation of indexes for search improvement (inverted file).
3	**Linear structures** Arrays. Methods for sorting and searching in arrays. Lists (single and double linked, circular). Advantages and disadvantages of arrays and lists for implementing indexes. Handling of updateable Iterators. Arrival-ordered data collections: Stack and Queue. Applications: expressions evaluator.
4	**Hashing** Hashing table. Hashing technique with collision handling (open and closed hashing). Implementations. Applications: advantages and disadvantages of hashing tables to implement indexes.
5	**Trees** Binary tree, expressions and search trees. Balanced AVL trees, and red-black trees. Multipath trees: B-tree of order N. Applications: advantages and disadvantages of search trees to implement indexes.
6	**Graphs**

Implementations with adjacency matrix, adjacency list, and incidence matrix. Advantages and disadvantages of each in problem solving. Traversals without heuristics: DFS (backtracking), BFS (queue use). Iterators.

Applications.

No.	Description
7	**Heuristics** Problem analysis by exploring the search space. Heuristic search. Applications.

Laboratory assignments:

No.	Description
1	**Laboratory assignments:** For all topics, students carry out individual practical work in the laboratory with Java language implementations.

Required bibliography:

No.	Description
1	Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, Introduction to Algorithms, Third Edition, MIT Press, 2009. Eric Roberts. Programming Abstractions in C. Addison Wesley, 1997.

Additional bibliography:

No.	Description
1	No additional bibliography has been uploaded.

Professor in charge:	Gómez, Leticia
Head of Department:	Bolo, Mario Enrique