Course guides
270687 - ADSDB - Algorithms, Data Structures and Databases

Last modified: 12/07/2021

Unit in charge: Barcelona School of Informatics
Teaching unit: 723 - CS - Department of Computer Science.
747 - ESSI - Department of Service and Information System Engineering.

Degree: MASTER'S DEGREE IN INNOVATION AND RESEARCH IN INFORMATICS (Syllabus 2012). (Optional subject).

Academic year: 2021 **ECTS Credits:** 6.0 **Languages:** English

LECTURER

Coordinating lecturer: OSCAR ROMERO MORAL

Others: Primer quadrimestre:
ANNA QUERALT CALAFAT - 11
MARIA JOSEFINA SIERRA SANTIBAÑEZ - 11

PRIOR SKILLS

This course assumes basic competences in algorithms, data structures and databases. The course is structured to cope with different backgrounds and learning needs but basic knowledge on Computer Science principles is assumed: notions of computer architecture, basic programming constructs and data structures.

DEGREE COMPETENCES TO WHICH THE SUBJECT CONTRIBUTES

Generical:
CG1. Capability to apply the scientific method to study and analyse of phenomena and systems in any area of Computer Science, and in the conception, design and implementation of innovative and original solutions.
CG2. Capability to lead, plan and supervise multidisciplinary teams.

Transversal:
CTR3. TEAMWORK: Capacity of being able to work as a team member, either as a regular member or performing directive activities, in order to help the development of projects in a pragmatic manner and with sense of responsibility; capability to take into account the available resources.
CTR4. INFORMATION LITERACY: Capability to manage the acquisition, structuring, analysis and visualization of data and information in the area of informatics engineering, and critically assess the results of this effort.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.

Basic:
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

TEACHING METHODOLOGY

During the first 10 weeks, the students are divided in two tracks: one for students with a minor in computer science (track 1) and another one for students with a major in computer science (track 2).

Students in track 1 will study fundamental concepts in algorithms, data structures and databases. First, additional material to read, study and understand is provided. Students will have a large bank of exercises to practice their understanding on their own. Then, in the face-to-face lectures, the lecturer will solve doubts and go through representative exercises to guarantee a solid understanding. Also, for the most complex concepts, some regular teaching sessions will be scheduled.

Students in track2 will undertake a research project related to advanced topics of algorithms, data structures and databases within the context of data science. Relevant and critical problems such as data integration, entity resolution and data quality will be proposed. The student will have access to a description of the problem, some seminal research papers and reference tools in this matter.

The last 5 weeks of the semester is common for all students regardless of the track they followed during the first 10 weeks. During these weeks the course project takes place. Students must create a end-to-end system architecture to ingest, store, process, learn models and deploy such system for a realistic project with realistic data.

This course has a strong self-learning component. Complementing it, the lecturers will provide additional material, advice hours and complementary lectures and labs to guarantee a solid comprehension.

LEARNING OBJECTIVES OF THE SUBJECT

1.To analyse the cost of iterative and recursive algorithms
2.To review some simple data structures: stacks, queues, lists, and trees
3.To know, explain, design, analyse, compare and implement the main data structures and algorithms that can be used to implement priority queues
4.To know, explain, design, analyse, compare and implement the main data structures and algorithms that can be used to implement dictionaries
5.To know, explain, design, analyse, compare and implement the main data structures and algorithms that can be used to represent graphs and solve classic graph problems such as traversals, topological ordering and shortest paths
6.To know, understand, explain, analyse and compare some algorithm design techniques: greedy, divide and conquer, and dynamic programming
7.To be aware of the limits of computation: to understand the definitions of the P and NP classes, the concept of Polynomial-Time reduction, the notion of NP-Completeness, and to know some classic NP-complete problems
8.Describe what is a database and a database management system
9.Effectively use the standard Structured Query Language (SQL) to query relational databases
10.Explain the relational data model, including its data structures, the relational algebra and integrity constraints
11.Given a set of informational requirements, model the logic schema of a relational database
12.Identify the main objectives of a database management system query optimizer
13.Develop a quality realistic end-to-end system architecture for a data science project

STUDY LOAD

Type	Hours	Percentage
Hours small group	25,5	17.00
Guided activities	3,0	2.00
Self study	96,0	64.00
Hours large group	25,5	17.00

Total learning time: 150 h

CONTENTS

Basics of Analysis of Algorithms

Description:
Worst case, best case and average case cost analysis. Asymptotic order of growth notations: Big-O, Omega and Theta. Analysis of the cost of iterative and recursive algorithms.

Simple Data Structures: Review

Description:
Stacks, queues, lists and trees.

Priority Queues

Description:
Operations of priority queues. Implementations with heaps. Heapsort.

Dictionaries

Description:
Operations of dictionaries. Basic implementations: tables and lists. Advanced implementations: hash tables, binary search trees, and AVL trees.

Graphs

Description:
Representations: adjacency matrices, adjacency lists and implicit representations. Depth-first search (DFS). Breadth-first search (BFS). Topological sort. Algorithms for shortest paths. Algorithm for minimum spanning trees.

Algorithm design techniques

Description:
Greedy, divide and conquer, and dynamic programming.

Introduction to NP and Computational Intractability

Description:
Basic introduction to P and NP classes, Polynomial-Time reduction, and NP-completeness. Examples of classic NP-complete problems.

Introduction to databases and database management systems

Description:
Main concepts on databases and database management systems. Relational database management systems.

SQL: Data-definition language and data-manipulation language

Description:
Introduction to the SQL language

The relational model

Description:
Data structures and integrity constraints. Views.

The relational algebra

Description:
The relational algebra operators and how to build data pipes with them. Notion of semantic and syntactic optimization.

Logical design of relational databases

Description:
Normalization theory. Translating conceptual schemas into relational schemas.

Notions of physical design and physical database optimization

Description:
Notions of query optimizer, access plan and cost model

Data Science-related advanced topics

Description:
Students with a major in Computer Science will investigate on advanced topics specific for data science projects. For example, data quality, entity resolution, data integration, etc.

ACTIVITIES

Basics of Analysis of Algorithms

Description:
To compare the efficiency of different algorithms for solving the same problem and select the most appropriate one. To compute the cost of an algorithm in the worst, best and average cases. To understand the definitions of the asymptotic order of growth notations Big-O, Omega and Theta, and their usefulness in characterising algorithm efficiency in time and space.

Specific objectives:
1

Related competencies :
CG1. Capability to apply the scientific method to study and analyse of phenomena and systems in any area of Computer Science, and in the conception, design and implementation of innovative and original solutions.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.
CTR4. INFORMATION LITERACY: Capability to manage the acquisition, structuring, analysis and visualization of data and information in the area of informatics engineering, and critically assess the results of this effort.
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

Full-or-part-time: 3h
Theory classes: 1h
Laboratory classes: 2h

Review of simple data structures

Description:
Operations. Lists, Stacks, Queues, Trees

Specific objectives:
2

Related competencies :
CG1. Capability to apply the scientific method to study and analyse of phenomena and systems in any area of Computer Science, and in the conception, design and implementation of innovative and original solutions.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.
CTR4. INFORMATION LITERACY: Capability to manage the acquisition, structuring, analysis and visualization of data and information in the area of informatics engineering, and critically assess the results of this effort.
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

Full-or-part-time: 4h
Theory classes: 1h
Laboratory classes: 3h

Priority Queues

Description:
Operations of priority queues. Implementations with heaps. Heapsort.

Specific objectives:
3

Related competencies :
CG1. Capability to apply the scientific method to study and analyse of phenomena and systems in any area of Computer Science, and in the conception, design and implementation of innovative and original solutions.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.
CTR4. INFORMATION LITERACY: Capability to manage the acquisition, structuring, analysis and visualization of data and information in the area of informatics engineering, and critically assess the results of this effort.
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

Full-or-part-time: 4h
Theory classes: 1h
Laboratory classes: 3h

Dictionaries

Description:
Operations of dictionaries and ordered dictionaries. Basic implementations: tables and lists. Advanced implementations: hash tables, binary search trees, AVL trees

Specific objectives:
4

Related competencies :
CG1. Capability to apply the scientific method to study and analyse of phenomena and systems in any area of Computer Science, and in the conception, design and implementation of innovative and original solutions.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.
CTR4. INFORMATION LITERACY: Capability to manage the acquisition, structuring, analysis and visualization of data and information in the area of informatics engineering, and critically assess the results of this effort.
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

Full-or-part-time: 4h
Theory classes: 1h
Laboratory classes: 3h

Graphs

Description:
Representations: adjacency matrices, adjacency lists and implicit representation. Depth-first search (DFS). Breadth-first search (BFS). Topological sort. Dijkstra's algorithm for shortest paths. Prim's algorithm for minimum spanning trees.

Specific objectives:
5

Related competencies :
CG1. Capability to apply the scientific method to study and analyse of phenomena and systems in any area of Computer Science, and in the conception, design and implementation of innovative and original solutions.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.
CTR4. INFORMATION LITERACY: Capability to manage the acquisition, structuring, analysis and visualization of data and information in the area of informatics engineering, and critically assess the results of this effort.
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

Full-or-part-time: 4h
Theory classes: 1h
Laboratory classes: 3h

Algorithmic schemes

Description:
Divide and conquer, Greedy algorithms, Dynamic Programming, Exhaustive search, Backtracking.

Specific objectives:
6

Related competencies :
CG1. Capability to apply the scientific method to study and analyse of phenomena and systems in any area of Computer Science, and in the conception, design and implementation of innovative and original solutions.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.
CTR4. INFORMATION LITERACY: Capability to manage the acquisition, structuring, analysis and visualization of data and information in the area of informatics engineering, and critically assess the results of this effort.
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

Full-or-part-time: 5h
Theory classes: 1h
Laboratory classes: 4h

Notions of Intractability

Description:
Basic introduction to P and NP classes. NP-completeness.

Specific objectives:
7

Related competencies :
CG1. Capability to apply the scientific method to study and analyse of phenomena and systems in any area of Computer Science, and in the conception, design and implementation of innovative and original solutions.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.
CTR4. INFORMATION LITERACY: Capability to manage the acquisition, structuring, analysis and visualization of data and information in the area of informatics engineering, and critically assess the results of this effort.
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

Partial Exam

Description:
For students with a minor in Computer Science, this exams evaluates their knowledge on fundamental concepts of algorithms, data structures and databases

Specific objectives:
1, 2, 3, 4, 5, 6, 7

Related competencies :
CG1. Capability to apply the scientific method to study and analyse of phenomena and systems in any area of Computer Science, and in the conception, design and implementation of innovative and original solutions.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.
CTR4. INFORMATION LITERACY: Capability to manage the acquisition, structuring, analysis and visualization of data and information in the area of informatics engineering, and critically assess the results of this effort.
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

Full-or-part-time: 8h
Guided activities: 2h
Self study: 6h

Introduction to databases and database management systems

Description:
The student attends the lecture, takes notes and participates in the session exercises

Specific objectives:
8

Related competencies :
CG1. Capability to apply the scientific method to study and analyse of phenomena and systems in any area of Computer Science, and in the conception, design and implementation of innovative and original solutions.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.
CTR4. INFORMATION LITERACY: Capability to manage the acquisition, structuring, analysis and visualization of data and information in the area of informatics engineering, and critically assess the results of this effort.
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

Full-or-part-time: 1h
Theory classes: 1h

SQL

Description:
The student attends the lecture, takes notes and participates in the session exercises

Specific objectives:
9

Related competencies :
CG1. Capability to apply the scientific method to study and analyse of phenomena and systems in any area of Computer Science, and in the conception, design and implementation of innovative and original solutions.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.
CTR4. INFORMATION LITERACY: Capability to manage the acquisition, structuring, analysis and visualization of data and information in the area of informatics engineering, and critically assess the results of this effort.
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

Full-or-part-time: 3h
Laboratory classes: 3h

The Relational Model

Description:
The student attends the lecture, takes notes and participates in the session exercises

Specific objectives:
10, 11

Related competencies :
CG1. Capability to apply the scientific method to study and analyse of phenomena and systems in any area of Computer Science, and in the conception, design and implementation of innovative and original solutions.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.
CTR4. INFORMATION LITERACY: Capability to manage the acquisition, structuring, analysis and visualization of data and information in the area of informatics engineering, and critically assess the results of this effort.
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

Full-or-part-time: 3h
Theory classes: 1h
Laboratory classes: 2h

Logical Design of Relational Databases

Description:
The student attends the lecture, takes notes and participates in the session exercises

Specific objectives:
10, 11

Related competencies :
CG1. Capability to apply the scientific method to study and analyse of phenomena and systems in any area of Computer Science, and in the conception, design and implementation of innovative and original solutions.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.
CTR4. INFORMATION LITERACY: Capability to manage the acquisition, structuring, analysis and visualization of data and information in the area of informatics engineering, and critically assess the results of this effort.
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

Full-or-part-time: 4h
Theory classes: 1h
Laboratory classes: 3h

Physical Optimization

Description:
The student attends the lecture, takes notes and participates in the session exercises

Specific objectives:
12

Related competencies :
CG1. Capability to apply the scientific method to study and analyse of phenomena and systems in any area of Computer Science, and in the conception, design and implementation of innovative and original solutions.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.
CTR4. INFORMATION LITERACY: Capability to manage the acquisition, structuring, analysis and visualization of data and information in the area of informatics engineering, and critically assess the results of this effort.
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

Full-or-part-time: 7h
Theory classes: 3h
Laboratory classes: 4h

Final Exam

Description:
For students with a minor in Computer Science, this exams evaluates their knowledge on fundamental concepts of algorithms, data structures and databases

Specific objectives:
8, 9, 10, 11, 12, 13

Related competencies :
CG1. Capability to apply the scientific method to study and analyse of phenomena and systems in any area of Computer Science, and in the conception, design and implementation of innovative and original solutions.
CG2. Capability to lead, plan and supervise multidisciplinary teams.
CTR3. TEAMWORK: Capacity of being able to work as a team member, either as a regular member or performing directive activities, in order to help the development of projects in a pragmatic manner and with sense of responsibility; capability to take into account the available resources.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.
CTR4. INFORMATION LITERACY: Capability to manage the acquisition, structuring, analysis and visualization of data and information in the area of informatics engineering, and critically assess the results of this effort.
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

Full-or-part-time: 7h
Guided activities: 2h
Self study: 5h

Data Science Advanced Topics project (DS-AT project)

Description:
Students with a major in Computer Science will investigate on advanced topics specific for data science projects. For example, data quality, entity resolution, data integration, etc.

Students with a minor in Computer Science will investigate and further study the fundamental concepts introduced in the lecturing hours.

Full-or-part-time: 44h
Laboratory classes: 4h
Self study: 40h

Data Science End-to-End Project (DS-EE)

Description:
All students will undertake a project spanning all main phases of a data science. As result, they are asked to develop a quality realistic end-to-end system architecture for a data science project.

Full-or-part-time: 49h
Laboratory classes: 4h
Self study: 45h

GRADING SYSTEM

Let E1 be the score of the partial exam,
E2 the score of the final exam,
RPM the score of the DS-AT project and
CPM the score of the DS-EE project.

Then,

NE = MAX(E2, E1)
If the student followed track 1 (see methodology) then NT = NE
else if student followed track 2 (see methodology) then NT = RPM

The final mark will be 0.6*NT + 0.4*CPM
Transversal competences will weight a 0% in the final score.

BIBLIOGRAPHY

Basic:
- Cormen, T.H. [et al.]. Introduction to algorithms [on line]. 4th ed. Cambridge: MIT Press, 2022 [Consultation: 11/04/2022]. Available on: https://search-ebscohost-com.recursos.biblioteca.upc.edu/login.aspx?direct=true&AuthType=ip,uid&db=nlebk&AN=2932690&site=ehost-live&ebv=EK&ppid=Page-__-1. ISBN 9780262046305.
- Goodrich, M.T.; Tamassia, R.; Goldwasser, M.H. Data structures and algorithm in Python. 1st ed. Wiley, 2013. ISBN 9781118290279.
- Garcia-Molina, H.; Ullman, J.D.; Widom, J. Database systems: the complete book [on line]. 2nd ed. Harlow, Essex: Pearson Education Limited, 2013 [Consultation: 16/07/2021]. Available on: https://ebookcentral-proquest-com.recursos.biblioteca.upc.edu/lib/upcatalunya-ebooks/reader.action?docID=5174436. ISBN 9781292024479.
- Lee, K.D.; Hubbard, S. Data Structures and Algorithms with Python. 1st. ed.. Springer, 2014. ISBN 9783319130712.
- Garey, M.R.; Johnson, D.S. Computers and intractability: a guide to the theory of NP-Completeness. W.H. Freeman, 1979. ISBN 0716710447.
- Sistac, J.; Camps, R. Bases de dades. UOC, 2005. ISBN 8497883349.
- Sistac, J. (coord.). Tècniques avançades de bases de dades. EDIUOC, 2000. ISBN 8484291065.
- Melton, J.; Simon, A.R. SQL: 1999: understanding relational, language components. Morgan Kaufmann, 2002. ISBN 1558604561.
- Ramakrishnan, R.; Gehrke, J. Database management systems. 3rd ed. McGraw-Hill, 2003. ISBN 0071151109.
- Teorey, T.J. [et al.]. Database modeling and design: logical design. 5th ed. Morgan Kaufmann Publishers/Elsevier, 2011. ISBN 9780123820204.
- Lightstone, S.; Teorey, T.J.; Nadeau, T. Physical database design: the database professional's guide to exploiting indexes, views, storage, and more [on line]. Morgan Kaufmann Publishers, 2007 [Consultation: 16/07/2021]. Available on: https://www.sciencedirect.com/science/book/9780123693891. ISBN 9780123693891.

Complementary:
- Manber, U. Introduction to algorithms: a creative approach. Repr. with corr. Addison-Wesley, 1989. ISBN 0201120372.
- Gulutzan, P.; Pelzer, T. SQL-99 complete, really. R & D books, 1999. ISBN 0879305681.
- Melton, J.; Eisenberg, A. Understanding SQL and Java together: a guide to SQLJ, JDBC, and related technologies. Morgan Kaufmann Publishers, 2000. ISBN 1558605622.
- Liu, L.; Özsu, M.T. Encyclopedia of database systems [on line]. Springer, 2009 [Consultation: 16/07/2021]. Available on: https://link.springer.com/referencework/10.1007/978-0-387-39940-9. ISBN 9780387399409.
- Lewis, J. Cost-based oracle fundamentals. Apress, 2006. ISBN 9781590596364.

RESOURCES

Hyperlink:
- http://learnsql.fib.upc.edu
- https://jutge.org