270687 - ADSDB - Algorithms, Data Structures and Databases

Coordinating unit: 270 - FIB - Barcelona School of Informatics
Teaching unit: 723 - CS - Department of Computer Science
747 - ESSI - Department of Service and Information System Engineering

Academic year: 2019
Degree: MASTER'S DEGREE IN INNOVATION AND RESEARCH IN INFORMATICS (Syllabus 2012). (Teaching unit Optional)
ECTS credits: 6
Teaching languages: English

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.

Teaching methodology

During the master classes the lecturer provides material related to the topic and proposes exercises. During the labs the lecturer proposes material and exercises to be solved with the computer. For the algorithms and data structures part, the "Jutge" tool will be used. For the database part, the "Learn-SQL" tool will be used.

Due to the heterogeneous background of the students taking the course, and also due to the fact this is a reinforcement course on basic Computer Science concepts, the autonomous learning is extremely important in this course. The student will be provided with several exercise to be done at home (whenever possible, through "Jutge" or "Learn-SQL"). It is the student responsibility to adapt the autonomous workload to his / her background needs.

Learning objectives of the subject

1. Calculate the efficiency of iterative and recursive algorithms
2. Review of simple data structures and its implementation: lists, stacks, queues, trees.
3. Understand, explain, design, analyse, compare and implement the main data structures that can be used to implement priority queues (trees, heaps).
4. Understand, explain, design, analyse, compare and implement the main data structures that can be used to implement dictionaries (tables, sorted tables, lists, sorted lists, hash tables, binary search trees, AVL trees).
5. Understand, explain, design, analyse, compare and implement algorithms that solve classic graph problems such as traversals, topological sorts, shortest paths, etc.
6. Understand, explain, design, analyse, compare and implement algorithmic schemes using diverse techniques (divide and conquer, greedy, backtracking, etc)
7. Identify computational limits: understand the implications of the question "P = NP?", understand the statement of Cook-Levin's Theorem, and recognise and identify several 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. Identify the main objectives of a database management system query optimizer
12. Given a set of informational requirements, model the logic schema of a relational database
13. Put in practice the course content on algorithmics, data structures and databases to solve real-life problems
# Study load

<table>
<thead>
<tr>
<th>Total learning time: 150h</th>
<th>Theory classes: 48h</th>
<th>32.00%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Practical classes: 0h</td>
<td>0.00%</td>
</tr>
<tr>
<td></td>
<td>Laboratory classes: 0h</td>
<td>0.00%</td>
</tr>
<tr>
<td></td>
<td>Guided activities: 6h</td>
<td>4.00%</td>
</tr>
<tr>
<td></td>
<td>Self study: 96h</td>
<td>64.00%</td>
</tr>
</tbody>
</table>
## Content

### Basics of Analysis of Algorithms
- **Degree competences to which the content contributes:**
- **Description:** Cost in time and space. Worst case, best case and average case. Asymptotic notation. Analysis of the cost of iterative and recursive algorithms. Recurrences

### Simple Data Structures: Review
- **Degree competences to which the content contributes:**
- **Description:** Operations. Lists, Stacks, Queues, Trees

### Priority Queues
- **Degree competences to which the content contributes:**
- **Description:** Operations of priority queues. Implementations with heaps. Heapsort.

### Dictionaries
- **Degree competences to which the content contributes:**
- **Description:** Operations of dictionaries and ordered dictionaries. Basic implementations: tables and lists. Advanced implementations: hash tables, binary search trees, AVL trees

### Graphs
- **Degree competences to which the content contributes:**

### Algorithmic schemes
- **Degree competences to which the content contributes:**
Description:
Divide and conquer, Greedy algorithms, Dynamic Programming, Exhaustive search, Backtracking.

Notions of Intractability

Degree competences to which the content contributes:
Description:
Basic introduction to P and NP classes. NP-completeness.

Introduction to databases and database management systems

Degree competences to which the content contributes:
Description:
Main concepts on databases and database management systems. Relational database management systems.

SQL: Data-definition language and data-manipulation language

Degree competences to which the content contributes:
Description:
Introduction to the SQL language

The relational model

Degree competences to which the content contributes:
Description:
Data structures and integrity constraints. Views.

The relational algebra

Degree competences to which the content contributes:
Description:
The relational algebra operators and how to build data pipes with them. Notion of semantic and syntactic optimization.

Logical design of relational databases

Degree competences to which the content contributes:
### Notions of physical design and physical database optimization

**Degree competences to which the content contributes:**

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

**Description:**
Normalization theory. Translating conceptual schemas into relational schemas.
# Planning of activities

## Basics of Analysis of Algorithms

<table>
<thead>
<tr>
<th>Hours: 8h</th>
</tr>
</thead>
<tbody>
<tr>
<td>Theory classes: 2h</td>
</tr>
<tr>
<td>Practical classes: 0h</td>
</tr>
<tr>
<td>Laboratory classes: 2h</td>
</tr>
<tr>
<td>Guided activities: 0h</td>
</tr>
<tr>
<td>Self study: 4h</td>
</tr>
</tbody>
</table>

**Description:**
Cost in time and space. Worst case, best case and average case. Asymptotic notation. Analysis of the cost of iterative and recursive algorithms. Recurrences

**Specific objectives:**
1


## Review of simple data structures

<table>
<thead>
<tr>
<th>Hours: 11h</th>
</tr>
</thead>
<tbody>
<tr>
<td>Theory classes: 1h 30m</td>
</tr>
<tr>
<td>Practical classes: 0h</td>
</tr>
<tr>
<td>Laboratory classes: 1h 30m</td>
</tr>
<tr>
<td>Guided activities: 0h</td>
</tr>
<tr>
<td>Self study: 8h</td>
</tr>
</tbody>
</table>

**Description:**
Operations. Lists, Stacks, Queues, Trees

**Specific objectives:**
2


## Priority Queues

<table>
<thead>
<tr>
<th>Hours: 12h</th>
</tr>
</thead>
<tbody>
<tr>
<td>Theory classes: 2h</td>
</tr>
<tr>
<td>Practical classes: 0h</td>
</tr>
<tr>
<td>Laboratory classes: 2h</td>
</tr>
<tr>
<td>Guided activities: 0h</td>
</tr>
<tr>
<td>Self study: 8h</td>
</tr>
</tbody>
</table>

**Description:**

**Specific objectives:**
3


## Dictionaries

<table>
<thead>
<tr>
<th>Hours: 12h</th>
</tr>
</thead>
<tbody>
<tr>
<td>Theory classes: 2h</td>
</tr>
<tr>
<td>Practical classes: 0h</td>
</tr>
<tr>
<td>Laboratory classes: 2h</td>
</tr>
<tr>
<td>Guided activities: 0h</td>
</tr>
<tr>
<td>Self study: 8h</td>
</tr>
</tbody>
</table>
### Description:
Operations of dictionaries and ordered dictionaries. Basic implementations: tables and lists. Advanced implementations: hash tables, binary search trees, AVL trees

### Specific objectives:
4

### Graphs

**Hours:** 16h  
- Theory classes: 3h  
- Practical classes: 0h  
- Laboratory classes: 3h  
- Guided activities: 2h  
- Self study: 8h

**Description:**  

**Specific objectives:**
5

### Algorithmic schemes

**Hours:** 21h  
- Theory classes: 5h  
- Practical classes: 0h  
- Laboratory classes: 5h  
- Guided activities: 0h  
- Self study: 11h

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

**Specific objectives:**
6

### Notions of Intractability

**Hours:** 10h  
- Theory classes: 1h  
- Practical classes: 0h  
- Laboratory classes: 1h  
- Guided activities: 0h  
- Self study: 8h

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

**Specific objectives:**
7
## First Partial Exam

<table>
<thead>
<tr>
<th>Description:</th>
<th>Hours: 10h</th>
</tr>
</thead>
<tbody>
<tr>
<td>This exam will assess the first 2/3 of the total content of the course</td>
<td>Guided activities: 2h</td>
</tr>
<tr>
<td></td>
<td>Self study: 8h</td>
</tr>
</tbody>
</table>

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

## Introduction to databases and database management systems

<table>
<thead>
<tr>
<th>Description:</th>
<th>Hours: 3h</th>
</tr>
</thead>
<tbody>
<tr>
<td>The student attends the lecture, takes notes and participates in the session exercises</td>
<td>Theory classes: 1h</td>
</tr>
<tr>
<td></td>
<td>Practical classes: 0h</td>
</tr>
<tr>
<td></td>
<td>Laboratory classes: 0h</td>
</tr>
<tr>
<td></td>
<td>Guided activities: 0h</td>
</tr>
<tr>
<td></td>
<td>Self study: 2h</td>
</tr>
</tbody>
</table>

### Specific objectives:
- 8

## SQL

<table>
<thead>
<tr>
<th>Description:</th>
<th>Hours: 13h</th>
</tr>
</thead>
<tbody>
<tr>
<td>The student attends the lecture, takes notes and participates in the session exercises</td>
<td>Theory classes: 1h</td>
</tr>
<tr>
<td></td>
<td>Practical classes: 0h</td>
</tr>
<tr>
<td></td>
<td>Laboratory classes: 4h</td>
</tr>
<tr>
<td></td>
<td>Guided activities: 0h</td>
</tr>
<tr>
<td></td>
<td>Self study: 8h</td>
</tr>
</tbody>
</table>

### Specific objectives:
- 9

## The Relational Model

<table>
<thead>
<tr>
<th>Description:</th>
<th>Hours: 11h</th>
</tr>
</thead>
<tbody>
<tr>
<td>The student attends the lecture, takes notes and participates in the session exercises</td>
<td>Theory classes: 2h</td>
</tr>
<tr>
<td></td>
<td>Practical classes: 0h</td>
</tr>
<tr>
<td></td>
<td>Laboratory classes: 2h</td>
</tr>
<tr>
<td></td>
<td>Guided activities: 1h</td>
</tr>
<tr>
<td></td>
<td>Self study: 6h</td>
</tr>
</tbody>
</table>

### Specific objectives:
- 10, 12
| Logical Design of Relational Databases | Hours: 7h  
Theory classes: 1h  
Practical classes: 0h  
Laboratory classes: 2h  
Guided activities: 0h  
Self study: 4h |
|--------------------------------------|----------------------------------|
| **Description:**  
The student attends the lecture, takes notes and participates in the session exercises |
| **Specific objectives:**  
10, 12 |

| Physical Optimization | Hours: 13h  
Theory classes: 4h  
Practical classes: 0h  
Laboratory classes: 1h  
Guided activities: 0h  
Self study: 8h |
|------------------------|----------------------------------|
| **Description:**  
The student attends the lecture, takes notes and participates in the session exercises |
| **Specific objectives:**  
11 |

| Second Partial Exam | Hours: 7h  
Guided activities: 2h  
Self study: 5h |
|---------------------|----------------------------------|
| **Description:**  
Closed book exam on databases and database management systems |
| **Specific objectives:**  
8, 9, 10, 11, 12, 13 |
Qualification system

Let $P_1$ be the score of the first partial exam.
$P_2$ the score of the second partial exam
$EF$ the score of the final exam
$Pr_1$ the score of the first practical work
$Pr_2$ the score of the second practical work
Then,
$NT = \text{MAX}(EF, P_1 \times 2/3 + P_2 \times 1/3)$
$NP = Pr_1 \times 2/3 + Pr_2 \times 1/3$
So the final score will be $0.6 \times NT + 0.4 \times NP$
Transversal competences will weight a 0% in the final score.
270687 - ADSDB - Algorithms, Data Structures and Databases

Bibliography

Basic:


Complementary:


Others resources:

Hyperlink

https://jutge.org

https://learnsql.fib.upc.edu/moodle/