270610 - ADS - Advanced Data Structures

Coordinating unit: 270 - FIB - Barcelona School of Informatics
Teaching unit: 723 - CS - Department of Computer Science
Academic year: 2018
Degree: MASTER'S DEGREE IN INNOVATION AND RESEARCH IN INFORMATICS (Syllabus 2012). (Teaching unit Optional)
ECTS credits: 6  Teaching languages: Catalan

Prior skills

Basic knowledge of the C++ programming language.
Basic knowledge of algorithm analysis methods (in particular asymptotic complexity).
Basic knowledge of elementary data structures such as stacks, queues, linked lists, trees, and graphs as well as of sorting methods such as insertion sort, heap sort, merge sort, and quick sort.

Degree competences to which the subject contributes

Basic:
CB6. Ability to apply the acquired knowledge and capacity for solving problems in new or unknown environments within broader (or multidisciplinary) contexts related to their area of study.
CB8. Capability to communicate their conclusions, and the knowledge and rationale underpinning these, to both skilled and unskilled public in a clear and unambiguous way.
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

Specific:
CEE3.1. Capability to identify computational barriers and to analyze the complexity of computational problems in different areas of science and technology as well as to represent high complexity problems in mathematical structures which can be treated effectively with algorithmic schemes.
CEE3.2. Capability to use a wide and varied spectrum of algorithmic resources to solve high difficulty algorithmic problems.

Generic:
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.
CG3. Capacity for mathematical modeling, calculation and experimental designing in technology and companies engineering centers, particularly in research and innovation in all areas of Computer Science.

Transversal:
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.
270610 - ADS - Advanced Data Structures

Teaching methodology

There will be two kinds of classes: theoretical sessions and practical sessions. On average, two hours a week is dedicated to theory and one hour a week to exercises. The lecturer will allocate the hours in accordance with the subject matter.

The theory classes take the form of lectures in which the lecturer sets out new concepts or techniques and examples illustrating them. Sessions will consist of a presentation of the main topics of each content’s item, mainly based in selected original research papers. A high level of students’ participation is expected at each session. Current lines of research in each topic will be discussed at the end of each topics' presentation.

The practical classes are used to carry out exercises in which students take an active part. Lecturers set the exercises in advance. Students are required to submit the exercises and then discuss the various solutions/alternatives in class.

Learning objectives of the subject

1. Become acquainted with the main and classic data structures of central areas of computer science and identify their major properties.
2. Become familiar with the mathematical tools usually used to analyze the performance of data structures.
3. Examine ideas, analysis and implementation details of data structures in order to assess their fitness to different classes of problems.
4. Select, design and implement appropriate data structures to solve given problems.

Study load

<table>
<thead>
<tr>
<th>Total learning time: 150h</th>
<th>Theory classes: 42h</th>
<th>28.00%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Practical classes: 12h</td>
<td>8.00%</td>
</tr>
<tr>
<td></td>
<td>Laboratory classes: 0h</td>
<td>0.00%</td>
</tr>
<tr>
<td></td>
<td>Guided activities: 0h</td>
<td>0.00%</td>
</tr>
<tr>
<td></td>
<td>Self study: 96h</td>
<td>64.00%</td>
</tr>
</tbody>
</table>
Content

Preliminaries.

**Degree competences to which the content contributes:**
**Description:**
Review of required previous knowledge: asymptotic notation, basic algorithm analysis, arrays, linked lists, stacks and queues, basics of hashing, binary search trees, AVL trees, red-black trees, heaps.

Hashing.

**Degree competences to which the content contributes:**
**Description:**
Hashing: Universal Hashing (construction of hash functions), Cuckoo Hashing (collision resolution strategies), Applications (Bloom Filters).

Heaps.

**Degree competences to which the content contributes:**
**Description:**
Heaps: Binomial Heaps.

Self-adjusting data structures.

**Degree competences to which the content contributes:**
**Description:**
Self-adjusting data structures: List updates, Splay trees.

Randomized data structures.

**Degree competences to which the content contributes:**
**Description:**
Randomized data structures: randomized BSTs, treaps.

Multidimensional and metric data structures, searching in metric spaces, associative retrieval and object representation.

**Degree competences to which the content contributes:**
**Description:**
Multidimensional and metric data structures, searching in metric spaces, associative retrieval and object representation: grid files, kd trees, point quad trees, PR quad trees, octrees.
### Geometric and kinetic data structures.

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

**Description:**
Geometric and kinetic data structures: interval, segment and partition trees, sweep lines.
Data structures for points in motion.

### Strings.

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

**Description:**
Strings: tries, Patricia tries, suffix trees, suffix arrays, BW-transform, FM-index

### External memory / cache oblivious.

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

**Description:**
External memory / cache oblivious: models, B-trees, ordered-file maintenance, van Emde-Boas layout.

### Succinct Data Structures.

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

**Description:**
Succinct rank and select operations.

### Miscellaneous.

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

**Description:**
Miscellaneous: concurrent, distributed, augmented, persistent data structures.
Qualification system

Grade = 60% FW + 20% FT + 10% SP + 10% Q

FW = Final Work (graded from 0 to 10) in which each participant is required to present a research paper (previously assigned by the lecturer). The presentation consists of:
- 3-5 minutes background on the topic of the paper, a motivation.
- 1 minute overview of the key ideas of the paper.
- 15 minutes presentation with most important details.
- 5 minutes demo of a program that implements the ideas introduced in the paper.

FT = Final test graded from (0 to 10) including all the contents of ADS.

SP = Summaries and participation (graded from 0 to 10) in which each participant is required to deliver a summary (1 page extent) of each others presentation and to participate (with questions and comments).

Q = Either quizzes (graded from 0 to 10), one per content's item or three (3) of the following deliverables:
- *Notes of one topic in latex (well explained and completed).
- *Read and resume one research paper.
- *Implement and prove experimentally one of the studied data structures.
- *Add to Wikipedia a data structure that isn't.

Bibliography

Basic:


