270611 - CPS - Combinatorial Problem Solving

Coordinating unit: 270 - FIB - Barcelona School of Informatics
Teaching unit: 723 - CS - Department of Computer Science
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

Basic knowledge on the Linux operating system and the C++ programming language.
Basic knowledge on linear algebra, graph algorithms and logics.

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.

Specific:
CEE3.2. Capability to use a wide and varied spectrum of algorithmic resources to solve high difficulty algorithmic problems.
CEE3.3. Capability to understand the computational requirements of problems from non-informatics disciplines and to make significant contributions in multidisciplinary teams that use computing.

General:
CG1. Capability to apply the scientific method to study and analyze 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:
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.
270611 - CPS - Combinatorial Problem Solving

Teaching methodology

The main feature of the teaching methodology is the use of materials accessible through the web, specifically designed for a self-learning course. These materials allow reformulating teaching in such a way that the traditional model of classes largely disappears.

Thus:

1. It regards the class as a baseline for work, which the student must continue and deepen on his/her own.

2. It builds upon high quality materials (slides, lists of problems, solved problems, examples of laboratory practical work, LP/SAT/CP software, bibliographic references).

3. It aims at motivating students, with examples, discussions, comments, etc... The intuitions behind the definitions, properties and techniques are discussed in group.

The laboratory will encourage independent work by the students. The role of the teacher will be mainly to assist and evaluate the students, who should work mostly autonomously.

Learning objectives of the subject

1. Modelling problems arising from computer science and other disciplines in the solving paradigms considered in the course: constraint programming, linear integer programming, propositional satisfiability.
2. Becoming familiar with state-of-the-art tools for the solving paradigms considered in the course: constraint programming, linear integer programming, propositional satisfiability.
3. Understanding the algorithmic foundations of each of the solving paradigms considered in the course: constraint programming, linear integer programming, propositional satisfiability.

Study load

<table>
<thead>
<tr>
<th>Total learning time: 141h</th>
<th>Hours large group: 30h</th>
<th>21.28%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Hours medium group:</td>
<td>0h</td>
</tr>
<tr>
<td></td>
<td>Hours small group:</td>
<td>15h</td>
</tr>
<tr>
<td></td>
<td>Guided activities:</td>
<td>0h</td>
</tr>
<tr>
<td></td>
<td>Self study:</td>
<td>96h</td>
</tr>
</tbody>
</table>
## Content

### Combinatorial Problems.

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

**Description:**
Informal definition. NP-complete problems vs. polynomial-time problems. Some examples and applications: propositional satisfiability, graph coloring, knapsack, bin packing, etc. Approaches to problem solving.

### Constraint Programming.

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

**Description:**

### Linear Programming.

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

**Description:**

### SAT solving and extensions.

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

**Description:**
# Planning of activities

<table>
<thead>
<tr>
<th>Course</th>
<th>Hours</th>
<th>Theory classes:</th>
<th>Practical classes:</th>
<th>Laboratory classes:</th>
<th>Guided activities:</th>
<th>Self study:</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Introduction to Combinatorial Problems</strong></td>
<td>4h 12m</td>
<td>2h</td>
<td>0h</td>
<td>0h</td>
<td>2h 12m</td>
<td>0h</td>
</tr>
<tr>
<td><strong>Constraint Programming</strong></td>
<td>29h</td>
<td>8h</td>
<td>0h</td>
<td>6h</td>
<td>2h 12m</td>
<td>12h 48m</td>
</tr>
<tr>
<td><strong>Linear Programming</strong></td>
<td>36h 48m</td>
<td>10h</td>
<td>0h</td>
<td>4h</td>
<td>2h 12m</td>
<td>20h 36m</td>
</tr>
<tr>
<td><strong>SAT and Extensions</strong></td>
<td>27h</td>
<td>8h</td>
<td>0h</td>
<td>4h</td>
<td>2h 12m</td>
<td>12h 48m</td>
</tr>
</tbody>
</table>

**Description:**
- Introduction to Combinatorial Problems
- Modelling and solving with Constraint Programming
- Modelling and solving with Linear Programming

**Specific objectives:**
1, 2, 3
### Description:
Modelling and solving with SAT and extensions

### Specific objectives:
1, 2, 3

| Final Exam | Hours: 21h 48m  
Guided activities: 2h  
Self study: 19h 48m |
|-------------|-------------------|
| Description:  
The exam covers the topics of modelling and solving with constraint programming, linear programming and propositional satisfiability. |
| Specific objectives:  
1, 2, 3 |

| Practical Work of Constraint Programming | Hours: 10h  
Guided activities: 0h  
Self study: 10h |
|------------------------------------------|-------------------|
| Description:  
The project consists in modelling and solving a combinatorial problem with constraint programming |
| Specific objectives:  
1, 2, 3 |

| Practical Work of Linear Programming | Hours: 10h  
Guided activities: 0h  
Self study: 10h |
|--------------------------------------|-------------------|
| Description:  
The project consists in modelling and solving a combinatorial problem with linear programming |
| Specific objectives:  
1, 2, 3 |

| Practical Work of SAT | Hours: 10h  
Guided activities: 0h  
Self study: 10h |
|-----------------------|-------------------|
| Description:  
The project consists in modelling and solving a combinatorial problem with SAT |
| Specific objectives:  
1, 2, 3 |
Qualification system

50% of the final grade corresponds to theory. This grade will be obtained by means of a written exam at the end of the course.

50% of the final grade corresponds to laboratory. This grade will be obtained as the mean of three successive projects (one for CP, another one for LP, and another one for SAT) that the students will have to hand in.

Bibliography

Basic:


Complementary:


Others resources:

Hyperlink

http://www.cs.upc.edu/~erodri/cps.html