
DEGREE CURRICULUM

# ADVANCED PROGRAMMING IN ARTIFICIAL INTELLIGENCE

Coordination: ARGELICH ROMA, JOSEP

Academic year 2017-18

# ADVANCED PROGRAMMING IN ARTIFICIAL INTELLI... 2017-18

## Subject's general information

| | |
|---|---|
| **Subject name** | ADVANCED PROGRAMMING IN ARTIFICIAL INTELLIGENCE |
| **Code** | 102038 |
| **Semester** | 2nd Q(SEMESTER) CONTINUED EVALUATION |
| **Typology** | <table><tr><th>Degree</th><th>Course</th><th>Typology</th><th>Modality</th></tr><tr><td>Bachelor's Degree in Computer Engineering</td><td>3</td><td>COMPULSORY</td><td>Attendance-based</td></tr></table> |
| **ECTS credits** | 6 |
| **Groups** | 1GG |
| **Theoretical credits** | 3 |
| **Practical credits** | 3 |
| **Coordination** | ARGELICH ROMA, JOSEP |
| **Department** | INFORMATICA I ENGINYERIA INDUSTRIAL |
| **Teaching load distribution between lectures and independent student work** | 6 ECTS = 25 x 6 = 150 hours<br><br>40% --> 60 hours of lectures<br>60% --> 90 hours of independent student work |
| **Important information on data processing** | Consult this link for more information. |
| **Language** | Catalan, English |
| **Office and hour of attention** | Please, send an email to the lecturer. |

# ADVANCED PROGRAMMING IN ARTIFICIAL INTELLI... 2017-18

| Teaching staff | E-mail addresses | Credits taught by teacher | Office and hour of attention |
|---|---|---|---|
| ARGELICH ROMA, JOSEP | jargelich@diei.udl.cat | 4,8 | Arrange meeting by e-mail. |
| PLANES CID, JORDI | jplanes@diei.udl.cat | 2,4 | |

## Subject's extra information

This course will delve into aspects of computing (Computational Science, ACM-IEEE CV-2008) and artificial intelligence (Intelligent Systems, ACM-IEEE CV-2008), guiding to an applied aspect, since each lesson will work with a different tool, solving problems.

To follow this subject properly some previous knowledge on computer programming, logics, and artificial intelligence are recommended.

## Learning objectives

- Understanding the different techniques and algorithms that are more commonly used in local search.
- Identifying problems where they can apply the local search techniques efficiently.
- Designing and implement local search algorithms for the Satisfiability and Maximum Satisfiability problems.
- Understanding how to evaluate the several implementations of algorithms to solve some problem in a neutral way.
- Knowing the different techniques and systematic search algorithms.
- Identifying problems which apply systematic search techniques.
- Designing and implement systematic search algorithms for Boolean satisfiability problem and maximum satisfiability .

## Competences

**Strategic Competences of the UdL**

**CT2.** Mastering a foreign language, especially English.
**CT3.** Training Experience in the use of the new technologies and the information and communication technologies.

**Cross-disciplinary competences**

**EPS6.** Capacity of analysis and synthesis.

**Specific competences**

**GII-C3.** Capacity to evaluate the computational complexity of a problem, to know the algorithmic strategies that can drive to its solving and recommend, develop and implement the one which guarantee the best performance in accordance with the requirements.
**GII-C4.** Capacity to understand the basics, paradigms and techniques of the intelligent systems and analyse, design and build systems, services and computer applications that use these techniques in any field of application.
**GII-C5.** Capacity to acquire, obtain, formalise and represent the human knowledge in a computable form to solve problems by means of a computer system in any field of application, particularly in the ones related with computation, perception and performance in environments or intelligent surroundings.

**GII-C6.** Capacity to develop and evaluate interactive systems and of presentation of complex information and its application to solve problems of design of computer-person interaction.
**GII-C7.** Capacity to know and develop techniques of computational learning and design and implement applications and systems that use them, including the ones devoted to automatic extraction of information and knowledge from big volumes of data.

## Subject contents

**1. Preliminaries**
1.1. Reminder SAT and MaxSAT

**2. The SAT problem**
2.1. Sistematic and non-sistematic algorithms
2.2. SAT local search
2.3. Neighborhood search
2.4. Genetic algorithms
2.5. Problem generation
2.6. SAT sistematic search

**3. The MaxSAT problem**
3.1. Branch and bound
3.2. Oracle-based solving

**4. Constraint programming**
4.1. Bucket elimination
4.2. Consistency
4.3. Optimization

## Methodology

**Large groups**

**Lectures:** exposition of subject contents orally by the teacher and with the help of notes and/or slides.

**Problems:** presentation of complex problems that the student tries to solve, followed by the solution proposed by the teacher.

**Practices:** application, on a practical level, of the contents given in the course.

**Written tests:** presential written test.

## Development plan

| Week | Description | Classroom Activity Big Group | Classroom/independent work |
|---|---|---|---|
| 1 | Lecture and problems | Lesson 1 | 4h/6h |
| 2 | Lecture and problems | Lesson 2.1 | 4h/6h |
| 3 | Lecture and problems | Lesson 2.2 | 4h/6h |
| 4 | Lecture and problems | Lesson 2.3 | 4h/6h |
| 5 | Lecture and problems | Lesson 2.4 | 4h/6h |

| Week | Description | Classroom Activity Big Group | Classroom/independent work |
|---|---|---|---|
| 6 | Lecture and problems | Lesson 2.5 | 4h/6h |
| 7 | Lecture and problems | Lesson 2.6 | 4h/6h |
| 8 | Practices | Presentation of practices | 4h/6h |
| 9 | Written tests | **First mid-term exam** | 2h/3h |
| 10 | Lecture and problems | Lesson 3 | 4h/6h |
| 11 | Lecture and problems | Lesson 3 | 4h/6h |
| 12 | Lecture and problems | Lesson 4 | 4h/6h |
| 13 | Lecture and problems | Lesson 4 | 4h/6h |
| 14 | Lecture and problems | Lesson 5 | 4h/6h |
| 15 | Practices | Presentation of practices | 4h/6h |
| 16 | Written tests | **Second mid-term exam** | 2h/3h |
| 17 | Written tests | **Second mid-term exam** | |
| 18 | | Study week | |
| 19 | Written tests | **Recovery exam** | |

## Evaluation

| Acornym | Evaluation activities | Weighting | Minimum score | Group work | Compulsory | Recoverable |
|---|---|---|---|---|---|---|
| P1 | Practice local search | 35% | 3 | Yes | Yes | Yes (*) |
| T1 | 1st mid-term Exam | 15% | 3 | No | Yes | Yes |
| P2 | Critical lecture | 10% | 3 | Yes | Yes | Yes (*) |
| P3 | Practice DPLL | 25% | 3 | Yes | Yes | Yes (*) |
| T2 | 2on mid-term Exam | 15% | 3 | No | Yes | Yes |
| (*) Recoverable activity with penalization | | | | | | |

**Final Score** = P1 * 0.35 + T1 * 0.15 + P2 * 0.1 + P3 * 0.25 + T2 * 0.15

In order to pass the course, the minimum grade for all activities and exams must be greater or equal to 3 (out of 10).

## Bibliography

**Basic bibliography**


- Armin Biere, Marijn Heule, Hans van Maaren, Toby Walsh (Eds.): Handbook of Satisfiability. Frontiers in Artificial Intelligence and Applications 185 IOS Press 2009
- Rina Dechter: Constraint processing. Elsevier Morgan Kaufmann 2003