# Course guide
# 270600 - AMMM - Algorithmic Methods for Mathematical Models

**Last modified:** 02/02/2024

**Unit in charge:** Barcelona School of Informatics
**Teaching unit:** 701 - DAC - Department of Computer Architecture.
723 - CS - Department of Computer Science.

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

**Academic year:** 2023 **ECTS Credits:** 6.0 **Languages:** English

## LECTURER

**Coordinating lecturer:** ENRIC RODRIGUEZ CARBONELL

**Others:** Primer quadrimestre:
ENRIC RODRIGUEZ CARBONELL - 11, 12
LUIS DOMINGO VELASCO ESTEBAN - 11, 12

Segon quadrimestre:
ENRIC RODRIGUEZ CARBONELL - 10
LUIS DOMINGO VELASCO ESTEBAN - 10

## PRIOR SKILLS

Students should be familiar with basic concepts in linear algebra: vector, matrix, rank, matrix inverse and solving systems of linear equations.

## DEGREE COMPETENCES TO WHICH THE SUBJECT CONTRIBUTES

**Specific:**
CEE2.1. Capability to understand models, problems and algorithms related to distributed systems, and to design and evaluate algorithms and systems that process the distribution problems and provide distributed services.
CEE3.2. Capability to use a wide and varied spectrum of algorithmic resources to solve high difficulty algorithmic problems.

**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.
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:**
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.

## TEACHING METHODOLOGY

Since the goal of the course is to provide the students with the necessary expertise to use different formalisms and tools to solve concrete problems, the teaching methodology will take that into account. Theory classes will always use motivating examples. In these sessions, students will solve simple exercises that will be key ingredients of more complicated algorithms.

In the laboratory sessions the students will become familiar with tools for solving problems computationally.

In the development of the project the students will be supervised by the instructors.

## LEARNING OBJECTIVES OF THE SUBJECT

1.Modelling in various mathematical formalisms the problems arising in different computer science disciplines
2.Becoming familiar with state-of-the-art tools used to solve various mathematical models
3.Understanding the basics of the algorithms used for solving various mathematical models

## STUDY LOAD

| Type | Hours | Percentage |
|---|---|---|
| Hours small group | 27,0 | 18.00 |
| Hours large group | 27,0 | 18.00 |
| Self study | 96,0 | 64.00 |

**Total learning time:** 150 h

## CONTENTS

### Linear Programming

**Description:**
Basics on linear programming. Modelling examples. The simplex algorithm. Duality.

### Integer linear programming

**Description:**
Modelling examples. Branch-and-bound, cuts and branch-and-cut.

### Non-linear programming

**Description:**
Basics on non-linear programming. Modelling examples.

### Metaheuristics

**Description:**
Constructive procedures. Local search. Metaheuristics: GRASP, Simulated Annealing, Tabu Search, Genetic algorithms, Ant Colony, Path Relinking, etc.

## ACTIVITIES

### Linear programming

**Specific objectives:**
1, 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.
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.
CEE3.2. Capability to use a wide and varied spectrum of algorithmic resources to solve high difficulty algorithmic problems.
CEE2.1. Capability to understand models, problems and algorithms related to distributed systems, and to design and evaluate algorithms and systems that process the distribution problems and provide distributed services.
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.

**Full-or-part-time:** 23h
Theory classes: 12h
Self study: 11h

### Integer Linear Programming

**Specific objectives:**
1, 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.
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.
CEE3.2. Capability to use a wide and varied spectrum of algorithmic resources to solve high difficulty algorithmic problems.
CEE2.1. Capability to understand models, problems and algorithms related to distributed systems, and to design and evaluate algorithms and systems that process the distribution problems and provide distributed services.
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.

**Full-or-part-time:** 20h
Theory classes: 8h
Self study: 12h

### Linear Programming Laboratory

**Specific objectives:**
2

**Related competencies :**
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.
CEE3.2. Capability to use a wide and varied spectrum of algorithmic resources to solve high difficulty algorithmic problems.
CEE2.1. Capability to understand models, problems and algorithms related to distributed systems, and to design and evaluate algorithms and systems that process the distribution problems and provide distributed services.
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.

**Full-or-part-time:** 13h
Laboratory classes: 4h
Self study: 9h

### Non-linear programming

**Specific objectives:**
1, 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.
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.
CEE3.2. Capability to use a wide and varied spectrum of algorithmic resources to solve high difficulty algorithmic problems.
CEE2.1. Capability to understand models, problems and algorithms related to distributed systems, and to design and evaluate algorithms and systems that process the distribution problems and provide distributed services.
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.

**Full-or-part-time:** 7h
Theory classes: 4h
Self study: 3h

### Metaheuristics

**Specific objectives:**
1, 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.
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.
CEE3.2. Capability to use a wide and varied spectrum of algorithmic resources to solve high difficulty algorithmic problems.
CEE2.1. Capability to understand models, problems and algorithms related to distributed systems, and to design and evaluate algorithms and systems that process the distribution problems and provide distributed services.
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.

**Full-or-part-time:** 28h
Theory classes: 16h
Self study: 12h

### Metaheuristics Laboratory

**Specific objectives:**
2

**Related competencies :**
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.
CEE3.2. Capability to use a wide and varied spectrum of algorithmic resources to solve high difficulty algorithmic problems.
CEE2.1. Capability to understand models, problems and algorithms related to distributed systems, and to design and evaluate algorithms and systems that process the distribution problems and provide distributed services.
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.

**Full-or-part-time:** 15h
Laboratory classes: 6h
Self study: 9h

**Project**

**Specific objectives:**
1, 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.
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.
CEE3.2. Capability to use a wide and varied spectrum of algorithmic resources to solve high difficulty algorithmic problems.
CEE2.1. Capability to understand models, problems and algorithms related to distributed systems, and to design and evaluate algorithms and systems that process the distribution problems and provide distributed services.
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.

**Full-or-part-time:** 27h
Guided activities: 3h
Self study: 24h

**Exam**

**Specific objectives:**
1, 2, 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.
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.
CEE3.2. Capability to use a wide and varied spectrum of algorithmic resources to solve high difficulty algorithmic problems.
CEE2.1. Capability to understand models, problems and algorithms related to distributed systems, and to design and evaluate algorithms and systems that process the distribution problems and provide distributed services.
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.

**Full-or-part-time:** 17h
Guided activities: 3h
Self study: 14h

## GRADING SYSTEM

The final grade of the course will take into account:

A) A practical work (project): 40%

B) A final exam: 60%

## BIBLIOGRAPHY

**Basic:**
- Luenberger, D.G.; Ye, Y. Linear and nonlinear programming. 4th ed. Springer, 2016. ISBN 9783319188416.
- Luenberger, D.G.; Ye, Y. Linear and nonlinear programming. 5th ed. Cham: Springer, 2021. ISBN 9783030854492.
- Avriel, M. Nonlinear programming: analysis and methods. Prentice-Hall, 1976. ISBN 0136236030.
- Glover, F.; Kochenberger, G.A. (eds.). Handbook of metaheuristics. Kluwer Academic Publishers, 2003. ISBN 1402072635.
- Ahuja, R.K.; Magnanti, T.L.; Orlin, J.B. Network flows: theory, algorithms, and applications. Pearson new int. ed. Harlow: Pearson, 2014. ISBN 9781292042701.
- Velasco, L.; Ruiz, M. Provisioning, recovery and in-operation planning in elastic optical network. John Wiley & Sons, Inc, 2017. ISBN 9781119338628.

**Complementary:**
- Williams, H.P. Model building in mathematical programming. 5th ed. Wiley & Sons, 2013. ISBN 9781118443330.
- Michalewicz, Z.; Fogel, D.B. How to solve it: modern heuristics. 2nd ed., rev. and ext. Springer, 2004. ISBN 3540224947.
- Larson, R. Elementary linear algebra. Eighth edition, metric version. Cengage Learning, 2017. ISBN 9781337556217.
- Bradley, S.P.; Hax, A.C.; Magnanti, T.L. Applied mathematical programming. Addison-Wesley, 1977. ISBN 020100464X.