Course guides
270407 - PA2 - Programming and Algorithms II

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

Degree: BACHELOR’S DEGREE IN ARTIFICIAL INTELLIGENCE (Syllabus 2021). (Compulsory subject).

Academic year: 2021   ECTS Credits: 6.0   Languages: Catalan

LECTURER

Coordinating lecturer: JORDI DELGADO PIN

Others: Segon quadrimestre:
JOSÉ LUIS BALCÁZAR NAVARRO - 11
JORDI DELGADO PIN - 11, 12

PRIOR SKILLS

PA1 course, or equivalent

DEGREE COMPETENCES TO WHICH THE SUBJECT CONTRIBUTES

Specific:
CE02. To master the basic concepts of discrete mathematics, logic, algorithmic and computational complexity, and its application to the automatic processing of information through computer systems. To be able to apply all these for solving problems.
CE03. To identify and apply the basic algorithmic procedures of computer technologies to design solutions to problems by analyzing the suitability and complexity of the proposed algorithms.
CE04. To design and use efficiently the most appropriate data types and structures to solve a problem.
CE10. To analyze, design, build and maintain applications in a robust, secure and efficient way, choosing the most appropriate paradigm and programming languages.
CE12. To master the fundamental principles and models of computing and to know how to apply them in order to interpret, select, assess, model, and create new concepts, theories, uses and technological developments related to artificial intelligence.
CE13. To evaluate the computational complexity of a problem, identify algorithmic strategies that can lead to its resolution and recommend, develop and implement the one that guarantees the best performance in accordance with the established requirements.

Generical:
CG2. To use the fundamental knowledge and solid work methodologies acquired during the studies to adapt to the new technological scenarios of the future.
CG4. Reasoning, analyzing reality and designing algorithms and formulations that model it. To identify problems and construct valid algorithmic or mathematical solutions, eventually new, integrating the necessary multidisciplinary knowledge, evaluating different alternatives with a critical spirit, justifying the decisions taken, interpreting and synthesizing the results in the context of the application domain and establishing methodological generalizations based on specific applications.
CG8. Perform an ethical exercise of the profession in all its facets, applying ethical criteria in the design of systems, algorithms, experiments, use of data, in accordance with the ethical systems recommended by national and international organizations, with special emphasis on security, robustness, privacy, transparency, traceability, prevention of bias (race, gender, religion, territory, etc.) and respect for human rights.

Transversal:
CT4. Teamwork. Be able to work as a member of an interdisciplinary team, either as a member or conducting management tasks, with the aim of contributing to develop projects with pragmatism and a sense of responsibility, taking commitments taking into account available resources.
CT6. Autonomous Learning. Detect deficiencies in one’s own knowledge and overcome them through critical reflection and the choice of the best action to extend this knowledge.
TEACHING METHODOLOGY

Teaching the course is structured in lectures and laboratory sessions.

Teachers will use lectures to introduce the essential contents of the course. In the laboratory sessions the contents of the course will be brought to the computer by carrying out practical problems. The laboratory classes will be a continuation of the lectures, where new concepts will be implemented as they appear in lectures.

LEARNING OBJECTIVES OF THE SUBJECT

1. Learn new data structures: Stacks, Queues, Lists, Trees and Graphs and the algorithms associated with the necessary operations on these structures.
2. Learn different implementations of data structures: Implementations using simpler data structures provided by the programming language, and dynamic implementations.
3. Learn the basics of object oriented programming: Concepts of class, instance, method, inheritance (single and multiple), etc.
4. Extension of Computational Complexity: Big O, Big Omega, Master Theorems
5. Solve a small/medium-sized problem in group, applying what they have learned about object orientation

STUDY LOAD

<table>
<thead>
<tr>
<th>Type</th>
<th>Hours</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Theory classes</td>
<td>30,0</td>
<td>20.00</td>
</tr>
<tr>
<td>Self study</td>
<td>90,0</td>
<td>60.00</td>
</tr>
<tr>
<td>Laboratory classes</td>
<td>30,0</td>
<td>20.00</td>
</tr>
</tbody>
</table>

Total learning time: 150 h

CONTENTS

Trees and Graphs

Description:
We will start the course using the data structures that the programming language provides us to implement Trees and Graphs. We will look at path algorithms for both structures, and other important algorithms related to these, such as Dijkstra's algorithm.

Objects and Classes

Description:
The fundamental concepts of object orientation are introduced: Class, object, instance, delegation, inheritance, etc. and other particularities of the implementation of object orientation specific to the programming language we work with.

Dynamic data structures

Description:
We will learn how to implement known data structures using the concept of reference to an object. We will re-implement data structures that the programming language provides by default, and new data structures that we have already seen implemented more easily.
Sets and Dictionaries: Implementation

Description:
We will look at different ways to implement sets and dictionaries: Hash Tables and Binary Search Trees (BSTs). The properties, advantages and disadvantages of these structures will be studied.

Algorithm Complexity (II)

Description:
In PA1 we started the study of the concept of complexity of a program, an algorithm and a problem, and we introduced the asymptotic notation, though only the definition of Theta (f). This topic is a continuation, where we will see the Big O, the Big Omega, etc., and the Master Theorems.

ACTIVITIES

Building Abstractions with Data: Trees and Graphs

Description:
The student should pay attention to the lecture and he/she should work through the exercises suggested by the lecturer.

Specific objectives:
1, 2

Related competencies:
CG2. To use the fundamental knowledge and solid work methodologies acquired during the studies to adapt to the new technological scenarios of the future.
CG4. Reasoning, analyzing reality and designing algorithms and formulations that model it. To identify problems and construct valid algorithmic or mathematical solutions, eventually new, integrating the necessary multidisciplinary knowledge, evaluating different alternatives with a critical spirit, justifying the decisions taken, interpreting and synthesizing the results in the context of the application domain and establishing methodological generalizations based on specific applications.
CE12. To master the fundamental principles and models of computing and to know how to apply them in order to interpret, select, assess, model, and create new concepts, theories, uses and technological developments related to artificial intelligence.
CE04. To design and use efficiently the most appropriate data types and structures to solve a problem.
CE03. To identify and apply the basic algorithmic procedures of computer technologies to design solutions to problems by analyzing the suitability and complexity of the proposed algorithms.
CE02. To master the basic concepts of discrete mathematics, logic, algorithmic and computational complexity, and its application to the automatic processing of information through computer systems. To be able to apply all these for solving problems.
CT6. Autonomous Learning. Detect deficiencies in one’s own knowledge and overcome them through critical reflection and the choice of the best action to extend this knowledge.

Full-or-part-time: 20h
Theory classes: 4h
Laboratory classes: 6h
Self study: 10h
Building Abstractions with Functions: Objects and Classes

Description:
The student should pay attention to the lecture and he/she should work through the exercises suggested by the lecturer.

Specific objectives:
2, 3

Related competencies:
CG2. To use the fundamental knowledge and solid work methodologies acquired during the studies to adapt to the new technological scenarios of the future.
CG4. Reasoning, analyzing reality and designing algorithms and formulations that model it. To identify problems and construct valid algorithmic or mathematical solutions, eventually new, integrating the necessary multidisciplinary knowledge, evaluating different alternatives with a critical spirit, justifying the decisions taken, interpreting and synthesizing the results in the context of the application domain and establishing methodological generalizations based on specific applications.
CE12. To master the fundamental principles and models of computing and to know how to apply them in order to interpret, select, assess, model, and create new concepts, theories, uses and technological developments related to artificial intelligence.
CE10. To analyze, design, build and maintain applications in a robust, secure and efficient way, choosing the most appropriate paradigm and programming languages.
CE04. To design and use efficiently the most appropriate data types and structures to solve a problem.
CE03. To identify and apply the basic algorithmic procedures of computer technologies to design solutions to problems by analyzing the suitability and complexity of the proposed algorithms.
CE02. To master the basic concepts of discrete mathematics, logic, algorithmic and computational complexity, and its application to the automatic processing of information through computer systems. To be able to apply all these for solving problems.
CT6. Autonomous Learning. Detect deficiencies in one's own knowledge and overcome them through critical reflection and the choice of the best action to extend this knowledge.

Full-or-part-time: 32h
Theory classes: 8h
Laboratory classes: 8h
Self study: 16h
Building Abstractions with Data: Dynamic Data Structures

Description:
The student should pay attention to the lecture and he/she should work through the exercises suggested by the lecturer.

Specific objectives:
1, 2

Related competencies:
CG2. To use the fundamental knowledge and solid work methodologies acquired during the studies to adapt to the new technological scenarios of the future.
CG4. Reasoning, analyzing reality and designing algorithms and formulations that model it. To identify problems and construct valid algorithmic or mathematical solutions, eventually new, integrating the necessary multidisciplinary knowledge, evaluating different alternatives with a critical spirit, justifying the decisions taken, interpreting and synthesizing the results in the context of the application domain and establishing methodological generalizations based on specific applications.
CE12. To master the fundamental principles and models of computing and to know how to apply them in order to interpret, select, assess, model, and create new concepts, theories, uses and technological developments related to artificial intelligence.
CE04. To design and use efficiently the most appropriate data types and structures to solve a problem.
CE03. To identify and apply the basic algorithmic procedures of computer technologies to design solutions to problems by analyzing the suitability and complexity of the proposed algorithms.
CE02. To master the basic concepts of discrete mathematics, logic, algorithmic and computational complexity, and its application to the automatic processing of information through computer systems. To be able to apply all these for solving problems.
CT6. Autonomous Learning. Detect deficiencies in one's own knowledge and overcome them through critical reflection and the choice of the best action to extend this knowledge.

Full-or-part-time: 20h
Theory classes: 6h
Laboratory classes: 4h
Self study: 10h
Sets and Dictionaries: Implementation

Description:
The student should pay attention to the lecture and he/she should work through the exercises suggested by the lecturer.

Specific objectives:
2, 3

Related competencies:
CG2. To use the fundamental knowledge and solid work methodologies acquired during the studies to adapt to the new technological scenarios of the future.
CG4. Reasoning, analyzing reality and designing algorithms and formulations that model it. To identify problems and construct valid algorithmic or mathematical solutions, eventually new, integrating the necessary multidisciplinary knowledge, evaluating different alternatives with a critical spirit, justifying the decisions taken, interpreting and synthesizing the results in the context of the application domain and establishing methodological generalizations based on specific applications.
CE12. To master the fundamental principles and models of computing and to know how to apply them in order to interpret, select, assess, model, and create new concepts, theories, uses and technological developments related to artificial intelligence.
CE10. To analyze, design, build and maintain applications in a robust, secure and efficient way, choosing the most appropriate paradigm and programming languages.
CE04. To design and use efficiently the most appropriate data types and structures to solve a problem.
CE03. To identify and apply the basic algorithmic procedures of computer technologies to design solutions to problems by analyzing the suitability and complexity of the proposed algorithms.
CE02. To master the basic concepts of discrete mathematics, logic, algorithmic and computational complexity, and its application to the automatic processing of information through computer systems. To be able to apply all these for solving problems.
CT6. Autonomous Learning. Detect deficiencies in one's own knowledge and overcome them through critical reflection and the choice of the best action to extend this knowledge.

Full-or-part-time: 16h
Theory classes: 4h
Laboratory classes: 4h
Self study: 8h
Algorithmic Complexity (II)

Description:
The student should pay attention to the lecture and he/she should work through the exercises suggested by the lecturer.

Specific objectives:
4

Related competencies:
CG2. To use the fundamental knowledge and solid work methodologies acquired during the studies to adapt to the new technological scenarios of the future.
CG4. Reasoning, analyzing reality and designing algorithms and formulations that model it. To identify problems and construct valid algorithmic or mathematical solutions, eventually new, integrating the necessary multidisciplinary knowledge, evaluating different alternatives with a critical spirit, justifying the decisions taken, interpreting and synthesizing the results in the context of the application domain and establishing methodological generalizations based on specific applications.
CE13. To evaluate the computational complexity of a problem, identify algorithmic strategies that can lead to its resolution and recommend, develop and implement the one that guarantees the best performance in accordance with the established requirements.
CE12. To master the fundamental principles and models of computing and to know how to apply them in order to interpret, select, assess, model, and create new concepts, theories, uses and technological developments related to artificial intelligence.
CE02. To master the basic concepts of discrete mathematics, logic, algorithmic and computational complexity, and its application to the automatic processing of information through computer systems. To be able to apply all these for solving problems.
CT6. Autonomous Learning. Detect deficiencies in one's own knowledge and overcome them through critical reflection and the choice of the best action to extend this knowledge.

Full-or-part-time: 16h
Theory classes: 4h
Laboratory classes: 4h
Self study: 8h
Practical Work

Description:
Students will do a pair practice related to a small-medium size problem.

Specific objectives:
1, 2, 3, 5

Related competencies:
CG2. To use the fundamental knowledge and solid work methodologies acquired during the studies to adapt to the new technological scenarios of the future.
CG4. Reasoning, analyzing reality and designing algorithms and formulations that model it. To identify problems and construct valid algorithmic or mathematical solutions, eventually new, integrating the necessary multidisciplinary knowledge, evaluating different alternatives with a critical spirit, justifying the decisions taken, interpreting and synthesizing the results in the context of the application domain and establishing methodological generalizations based on specific applications.
CG8. Perform an ethical exercise of the profession in all its facets, applying ethical criteria in the design of systems, algorithms, experiments, use of data, in accordance with the ethical systems recommended by national and international organizations, with special emphasis on security, robustness, privacy, transparency, traceability, prevention of bias (race, gender, religion, territory, etc.) and respect for human rights.
CE12. To master the fundamental principles and models of computing and to know how to apply them in order to interpret, select, assess, model, and create new concepts, theories, uses and technological developments related to artificial intelligence.
CE10. To analyze, design, build and maintain applications in a robust, secure and efficient way, choosing the most appropriate paradigm and programming languages.
CE04. To design and use efficiently the most appropriate data types and structures to solve a problem.
CE03. To identify and apply the basic algorithmic procedures of computer technologies to design solutions to problems by analyzing the suitability and complexity of the proposed algorithms.
CE02. To master the basic concepts of discrete mathematics, logic, algorithmic and computational complexity, and its application to the automatic processing of information through computer systems. To be able to apply all these for solving problems.
CT6. Autonomous Learning. Detect deficiencies in one's own knowledge and overcome them through critical reflection and the choice of the best action to extend this knowledge.
CT4. Teamwork. Be able to work as a member of an interdisciplinary team, either as a member or conducting management tasks, with the aim of contributing to develop projects with pragmatism and a sense of responsibility, taking commitments taking into account available resources.

Full-or-part-time: 12h
Guided activities: 2h
Self study: 10h
## Partial Exam

### Specific objectives:
1, 2, 3

### Related competencies:
- **CG2.** To use the fundamental knowledge and solid work methodologies acquired during the studies to adapt to the new technological scenarios of the future.
- **CG4.** Reasoning, analyzing reality and designing algorithms and formulations that model it. To identify problems and construct valid algorithmic or mathematical solutions, eventually new, integrating the necessary multidisciplinary knowledge, evaluating different alternatives with a critical spirit, justifying the decisions taken, interpreting and synthesizing the results in the context of the application domain and establishing methodological generalizations based on specific applications.
- **CE12.** To master the fundamental principles and models of computing and to know how to apply them in order to interpret, select, assess, model, and create new concepts, theories, uses and technological developments related to artificial intelligence.
- **CE10.** To analyze, design, build and maintain applications in a robust, secure and efficient way, choosing the most appropriate paradigm and programming languages.
- **CE04.** To design and use efficiently the most appropriate data types and structures to solve a problem.
- **CE03.** To identify and apply the basic algorithmic procedures of computer technologies to design solutions to problems by analyzing the suitability and complexity of the proposed algorithms.
- **CE02.** To master the basic concepts of discrete mathematics, logic, algorithmic and computational complexity, and its application to the automatic processing of information through computer systems. To be able to apply all these for solving problems.
- **CT6.** Autonomous Learning. Detect deficiencies in one’s own knowledge and overcome them through critical reflection and the choice of the best action to extend this knowledge.

### Full-or-part-time: 12h
- Guided activities: 2h
- Self study: 10h

## Final Exam

### Specific objectives:
1, 2, 3, 4

### Related competencies:
- **CG2.** To use the fundamental knowledge and solid work methodologies acquired during the studies to adapt to the new technological scenarios of the future.
- **CG4.** Reasoning, analyzing reality and designing algorithms and formulations that model it. To identify problems and construct valid algorithmic or mathematical solutions, eventually new, integrating the necessary multidisciplinary knowledge, evaluating different alternatives with a critical spirit, justifying the decisions taken, interpreting and synthesizing the results in the context of the application domain and establishing methodological generalizations based on specific applications.
- **CE13.** To evaluate the computational complexity of a problem, identify algorithmic strategies that can lead to its resolution and recommend, develop and implement the one that guarantees the best performance in accordance with the established requirements.
- **CE12.** To master the fundamental principles and models of computing and to know how to apply them in order to interpret, select, assess, model, and create new concepts, theories, uses and technological developments related to artificial intelligence.
- **CE10.** To analyze, design, build and maintain applications in a robust, secure and efficient way, choosing the most appropriate paradigm and programming languages.
- **CE04.** To design and use efficiently the most appropriate data types and structures to solve a problem.
- **CE03.** To identify and apply the basic algorithmic procedures of computer technologies to design solutions to problems by analyzing the suitability and complexity of the proposed algorithms.
- **CE02.** To master the basic concepts of discrete mathematics, logic, algorithmic and computational complexity, and its application to the automatic processing of information through computer systems. To be able to apply all these for solving problems.
- **CT6.** Autonomous Learning. Detect deficiencies in one’s own knowledge and overcome them through critical reflection and the choice of the best action to extend this knowledge.

### Full-or-part-time: 13h
- Guided activities: 3h
- Self study: 10h
**GRADING SYSTEM**

Grading the course will consist of two theoretical tests (T1 and T2), one mid-course and the other at the end, and one medium-sized practical work (Practice).

Then, the evaluation method would be:

0.6 * Theory + 0.4*Practice  
where:  
Theory: MAX(T2, 0.5* T1 + 0.5* T2)

Teamwork:

Evaluated using a simple rubric that each group tutor group uses to rank different aspects of teamwork of every member of the group.

**BIBLIOGRAPHY**

**Basic:**
- DeNero, John. Composing Programs.

**Complementary:**

**RESOURCES**

**Hyperlink:**
- [https://docs.python.org/3/reference/index.html](https://docs.python.org/3/reference/index.html)