Course guide
340374 - ESIN-I3O23 - Information Structure

Last modified: 17/05/2023

Unit in charge: Vilanova i la Geltrú School of Engineering
Teaching unit: 723 - CS - Department of Computer Science.

Degree: BACHELOR'S DEGREE IN INFORMATICS ENGINEERING (Syllabus 2018). (Compulsory subject).

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

LECTURER

Coordinating lecturer: Bernardino Casas Fernández, Jordi Esteve Cusiné

Others: Jordi Esteve Cusiné

PRIOR SKILLS

Proficiency of imperative object-based programming techniques:
* Classes
* Objects
* Methods
* How to pass parameters
* Recursion

Know well at least one object-oriented imperative language, preferably C++.

Mathematical and algorithmic maturity.

REQUIREMENTS

Have passed PRO1 or at least have the knowledge taught in PRO1.

DEGREE COMPETENCES TO WHICH THE SUBJECT CONTRIBUTES

Specific:
1. CEFB3. Ability to understand and to have a good command of discrete, logical, algorithmically mathematics and computing complexity and its application to automatical treatment of information by means of computational systems and its application to solve engineering problems.
2. CEFB4. Basic knowledge of use and computer programming, as well as of operating systems, data base and generally informatic programs with engineering applications.
3. CEFB5. Knowledge of informatic systems, its structure, function and interconnection, as well as fundamentals of its programming.
4. CEFC6. Basic knowledge and application of algorithmic processes, informatic techniques to design solutions of problems, analyzing if proposed algorisms are apt and complex.
5. CEFC7. Knowledge, design and efficient use of data types and structures the most appropriate to resolve problems.
6. CEFC8. Ability to analyze, to design, to construct and to maintain applications in a well built, secure and efficient way choosing the most adequated paradigms and languages.

Transversal:
7. SELF-DIRECTED LEARNING - Level 2: Completing set tasks based on the guidelines set by lecturers. Devoting the time needed to complete each task, including personal contributions and expanding on the recommended information sources.
8. TEAMWORK - Level 1. Working in a team and making positive contributions once the aims and group and individual responsibilities have been defined. Reaching joint decisions on the strategy to be followed.

TEACHING METHODOLOGY

The methodological approach consists in:
- Presentation in classroom, in participatory classes, of concepts and procedures associated with the subjects.
- Individually or in team exercises inside and outside the classroom.
- Completion of individually practices (activities) or in a team practices (project) inside and outside the classroom.
- Individual study, tests and exams.

LEARNING OBJECTIVES OF THE SUBJECT

Provide the student with the ability to specify, design, implement and evaluate data structures and the ability to identify the most appropriate algorithms on these structures. Likewise, it is intended to provide the student with more experience in the field of programming by carrying out activities and a project.

1. Specify: The student must learn some informal specification techniques that will allow him to describe the behavior of the classes.

2. Design: The student must learn to select and combine algorithmic techniques and appropriate data structures to develop modular designs for medium difficulty problems.

3. Implement: The student will learn the main data organization techniques that allow accessing and modifying information efficiently, respecting the corresponding specification.

4. Evaluate: The student will be able to compare the relative efficiency of alternative solutions to the same problem and make design decisions based on objective criteria based on efficiency in execution time and memory space used.

STUDY LOAD

Type	Hours	Percentage
Self study	90,0	60.00
Hours large group	30,0	20.00
Hours small group	30,0	20.00

Total learning time: 150 h

CONTENTS

1. Object oriented programming

Description:
Class specification.
Classes and objects.
Attributes and methods.
Parameter passing.
Generalization.
Method overload.

IMPORTANT: This topic is for review, therefore there is no face-to-face teaching of it.

Full-or-part-time: 5h
Self study : 5h

2. Analysis of the efficiency of algorithms

Description:
Measurements of the cost in space and time of algorithms. Asymptotic notations. Cost in worst and average cases. Rules for calculating the efficiency of iterative and recursive algorithms.

Related activities:
Activitat 1
Activitat 2
Activitat 3
Control 1
Control 2
Projecte

Full-or-part-time: 7h
Theory classes: 2h
Self study : 5h

3. Static linear structures

Description:
Review of the concept of sequence.
Stacks and queues.
Lists.
Applications.
Vector and linked implementation of linear structures.

IMPORTANT: This topic is for review, therefore the face-to-face teaching of it will be very brief.

Full-or-part-time: 8h
Theory classes: 1h
Laboratory classes: 2h
Self study : 5h

4. Dynamic linear structures

Description:
Pointers.
Obtain and free memory.
Stacks, queues, and linked lists with dynamic memory.
Iterators.
Merge sorting (mergesort).

Related activities:
Activitat 1
Control 1
Control 2
Projecte

Full-or-part-time: 22h
Theory classes: 4h
Laboratory classes: 6h
Self study : 12h

5. Trees

Description:
Review of the tree concept.
General trees.
Binary trees.
Applications.
Tree transversals in preorder, postorder, inorder and by levels.
Implementation of trees with vector of pointers to children.
Tree implementation with first-child / next-sibling pointers.

Related activities:
Control 1
Control 2
Activitat 2
Projecte

Full-or-part-time: 18h
Theory classes: 4h
Laboratory classes: 4h
Self study : 10h

6. Dictionaries

Description:
Specification of dictionaries.
Applications.
Simple implementation techniques: lists, ordered lists, self-organizing lists.
Advanced implementation techniques: binary search trees (BSTs), binary balanced search trees (AVLs), hash tables (Hash Tables), digital trees (Tries).
Quicksort and Radixsort algorithms.

Related activities:
Control 2
Activitat 3
Projecte

Full-or-part-time: 27h
Theory classes: 8h
Laboratory classes: 4h
Self study : 15h

7. Priority queues

Description:
Specification of priority queues.
Applications.
Implementation through Heaps.
Heapsort algorithm.

Related activities:
Control 2
Activitat 3
Projecte

Full-or-part-time: 7h
Theory classes: 2h
Laboratory classes: 1h
Self study : 4h

8. Graphs

Description:
Representations with adjacency matrices, adjacency lists, and adjacency multilists.
Depth-First Search traversal (DFS).
Breadth-First Search traversal (BFS).
Topological ordering.
Dijkstra's algorithm for minimal paths.
Kruskal's algorithm for minimal spanning trees.

Related activities:
Control 2

Full-or-part-time: 14h
Theory classes: 5h
Laboratory classes: 1h
Self study : 8h

Project

Description:
Programming project done in couples. The project will consist of the implementation of a modular design with the C ++ language. The student's task will be to implement different modules of the modular design given in the statement. The project statement and the ESIN scheduling guide and standards will be available.

Full-or-part-time: 42h
Laboratory classes: 8h
Guided activities: 34h

ACTIVITIES

Activity 1

Description:
See Catalan version

Material:
See Catalan version.

Delivery:
See Catalan version.

Full-or-part-time: 3h 30m
Laboratory classes: 3h
Guided activities: 0h 30m

Activity 2

Description:
See Catalan version

Material:
See Catalan version.

Delivery:
See Catalan version

Full-or-part-time: 3h 30m
Laboratory classes: 3h
Guided activities: 0h 30m

Activity 3

Description:
See catalan version.

Material:
See catalan version.

Delivery:
See catalan version.

Full-or-part-time: 3h 30m
Laboratory classes: 3h
Guided activities: 0h 30m

Control 1

Description:
See Catalan version.

Delivery:
See Catalan version.

Full-or-part-time: 2h
Theory classes: 2h

Control 2

Description:
See Catalan version.

Delivery:
See Catalan version.

Full-or-part-time: 3h
Theory classes: 3h

Project

Description:
See catalan version.

Material:
See catalan version.

Delivery:
See catalan version.

Full-or-part-time: 27h
Laboratory classes: 7h
Self study: 20h

GRADING SYSTEM

C1 = Control 1 (partial test). Individual test (2 hours).
C2 = Control 2 (final test). Individual test (3 hours) which integrates knowledge and skills of the entire course.
Act = Result of the 3 performed activities.
Pro = Result of the project.

Theory Qualification: QT = maximum value (0.40*C1+0.60*C2, C2)

Final Qualification: QF = 0.5*QT + 0.2*Act + 0.3*Pro

Project (Pro): It is evaluated from: the execution of the program, the delivered code, and a Validation Test (PV). The Validation Test can be done through a face-to-face interview, or in the Final Control of the subject.

The realization and presentation of the project will be a necessary condition for passing the subject. Otherwise, the final grade for the entire subject will be "Not Presented".

You can reevaluate Control 2 (final test).

EXAMINATION RULES.

Written tests (Control 1 and 2) and Activities are on-site and individual, mainly done on jutge.org platform.

The project is carried out in teams of two people. It is delivered in a non-contact manner and is evaluated both face-to-face (Validation Trial) and non-contact from the documentation submitted.

BIBLIOGRAPHY

Basic:

- Bernardino Casas, Jordi Esteve. Apunts d'ESIN. Transparències pels alumnes. Campus virtual,
- Bernardino Casas, Jordi Esteve. Col·lecció de problemes d'ESIN. Campus virtual,
- Bernardino Casas, Jordi Esteve. Manual de laboratori d'ESIN. Campus virtual,
- Hernández Orallo, Enrique; Hernández Orallo, José; Juan Lizandra, Mª Carmen. C++ estándar : [programación con el estándar ISO y la biblioteca de plantilla (STL)]. Madrid: Paraninfo, 2002. ISBN 8497320409.
- Guardati Buemo, Silvia. Estructura de datos orientada a objetos : algoritmos con C++ [on line]. Naucalpan de Juárez (México): Pearson, 2007 [Consultation: 16/02/2024]. Available on: https://www-ingebook-com.recursos.biblioteca.upc.edu/ib/NPcd/IB_BooksVis?cod_primaria=1000187&codigo_libro=3887. ISBN 9702607922.
- Martí Oliet, Narciso; Ortega Mallén, Yolanda; Verdejo López, José Alberto. Estructuras de datos y métodos algorítmicos. Ejercicios resueltos [on line]. Madrid [etc.]: Pearson Educación, 2001 [Consultation: 06/03/2024]. Available on: https://www-ingebook-com.recursos.biblioteca.upc.edu/ib/NPcd/IB_BooksVis?cod_primaria=1000187&codigo_libro=1316. ISBN 8420538493.

Complementary:

- Stroustrup, Bjarne. El lenguaje de programación C++. Madrid [etc.]: Addison Wesley, 2002 (reimpressió 2007). ISBN 847829046X.
- Joyanes Aguilar, Luis. Programación en C++ : algoritmos, estructuras de datos y objetos [on line]. 2a ed. Madrid: McGraw-Hill, 2006 [Consultation: 12/02/2024]. Available on: https://www-ingebook-com.recursos.biblioteca.upc.edu/ib/NPcd/IB_BooksVis?cod_primaria=1000187&codigo_libro=4071. ISBN 844814645X.