Course guide
310604 - 310604 - Informatics

Last modified: 07/11/2023

Unit in charge:	Barcelona School of Building Construction
Teaching unit:	751 - DECA - Department of Civil and Environmental Engineering.

Degree:	BACHELOR'S DEGREE IN GEOINFORMATION AND GEOMATICS ENGINEERING (Syllabus 2016). (Compulsory subject).

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

LECTURER

Coordinating lecturer: Albert Prades Valls

Others:

DEGREE COMPETENCES TO WHICH THE SUBJECT CONTRIBUTES

Specific:
1. Basic knowledge about use and computer programmation, operative systems, databases and software programmes with application in the enginnering.
2. (ENG) Determinar, mesurar, avaluar i representar el terreny, objectes tridimensionals, punts i trajectòries.
3. (ENG) Gestió i execució de projectes d'investigació, de desenvolupament i d'innovació dins l'àmbit d'aquesta enginyeria.
4. (ENG) Planificació, projecte, direcció, execució i gestió de processos de mesura, sistemes d'informació, explotació d'imatges, posicionament i navegació; modelització, representació i visualització de la informació territorial en, sota i sobre la superfície terrestre.
6. (ENG) Planificació, projecte, direcció, execució i gestió de processos i productes d'aplicació en cadastre i registre, ordenació del territori i valoració, dins l'àmbit geomàtic.
5. (ENG) Planificació, projecte, direcció, execució i gestió de processos i productes d'aplicació a la societat de l'informació dins l'àmbit geomàtic.

Transversal:
7. THIRD LANGUAGE. Learning a third language, preferably English, to a degree of oral and written fluency that fits in with the future needs of the graduates of each course.

TEACHING METHODOLOGY

The directed learning hours consist of giving theoretical classes (large group) in which the teacher makes a brief presentation to introduce the general learning objectives related to the basic concepts of the subject. Subsequently and through practical programming exercises, it tries to motivate and involve the students so that they actively participate in their learning.

Support material is used through ATENEA: learning objectives by content, concepts, examples, programming of evaluation and directed learning activities and bibliography. They also consist of teaching programming classes in which they work on solving exercises or problems related to the specific learning objectives of each of the contents of the subject.

In these practical sessions it is intended to incorporate some generic competences.

After each session, tasks outside the classroom are proposed, which must be worked on individually.

Other hours of autonomous learning must also be considered, such as those devoted to oriented reading and solving the problems proposed on the different contents, through the virtual campus ATENEA.

LEARNING OBJECTIVES OF THE SUBJECT

In this subject it is intended that the student reaches a minimum of solvency instead of thinking, designing and implementing programs to solve the different problems that arise in other subjects of the career.

STUDY LOAD

Type	Hours	Percentage
Hours large group	24,0	16.00
Hours medium group	36,0	24.00
Self study	90,0	60.00

Total learning time: 150 h

CONTENTS

C1 Introduction to C ++ Programming

Description:

-The course begins with the basic programming topics:

· Introduction to the C ++ language: identifiers, reserved words, preprocessor, data types, assignment operators, arithmetic, relational, logical, operator priority.

· Flow control: conditional constructions (if-else, switch) and iterative constructions (do-while, while, for), nesting.

· Functions and the structure of the program: definition, return of values, argument of a function, not by value and not by reference, recursive functions.

· Vectors, strings and pointers. Matrices and double pointers. Data structures.

Input and output for files (fopen, fclose, fwrite, fread, fprintf, fscanf)

· Dynamic memory allocation: runtime memory reservation (new, delete).

'Is not correct?

The course begins with the basic programming topics:
· Introduction to the C ++ language: identifiers, reserved words, preprocessor, data types, assignment operators, arithmetic, relational, logical, operator priority.
· Flow control: conditional constructions (if-else, switch) and iterative constructions (do-while, while, for), nesting.
· Functions and the structure of the program: definition, return of values, argument of a function, not by value and not by reference, recursive functions.
· Vectors, strings and pointers. Matrices and double pointers. Data structures.
Input and output for files (fopen, fclose, fwrite, fread, fprintf, fscanf)
· Dynamic memory allocation: runtime memory reservation (new, delete).

Related activities:
Management of the programming environment: Visualnet (activity1)
Small program programming practices (activity2).

Full-or-part-time: 38h
Theory classes: 6h
Practical classes: 9h
Self study : 23h

C2 Object Oriented Programming in C ++

Description:
Introduction to the definition of an object and properties:
· Classes, encapsulation, polymorphism and inheritance.
· Private, public and protected variables and members, friendly functions.

Basic applications:
· Queues, stacks, linked lists, trees.
· Sorting algorithms.
· Search algorithms.

Full-or-part-time: 38h
Theory classes: 6h
Practical classes: 9h
Self study : 23h

C3 Introduction to algorithm

Description:
1) Data structures: queues, stacks, linked lists, trees.
2) Sort methods: bubble, shell.
3) Search methods.
4) Std data structures.

Full-or-part-time: 37h
Theory classes: 6h
Practical classes: 9h
Self study : 22h

C4 Brief introduction to Python

Description:
Introduction to Python

Full-or-part-time: 37h
Theory classes: 6h
Practical classes: 9h
Self study : 22h

ACTIVITIES

ACTIVITY 1

Description:
Basic handling of the VisualStudio programming environment

Specific objectives:
At the end of the test, the student must be able to:

· Create a new project.
· Create header files and code as necessary.
· Compile, run, debug and correct where appropriate.

Material:
The documentation will be found in Athena. The exercise will be carried out in the calculation center.

Delivery:
The activity will take place during the second week and is not evaluated.

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

ACTIVITY 2

Description:
A collection of exercises (10-12) of basic programming will be solved to solve simple but repetitive exercises of the optimal type for a computer to do:
· Calculation of prime numbers from 1 to N.
· Calculation of the frequency of appearance of a letter or a word in a text.
· Etc.

Specific objectives:
At the end of the proposed exercises the student must be able to design and implement relatively simple programs. You must have learned to be minimally orderly in implementing an algorithm and to make use of interactive and flow constructs.

Material:
The documentation will be found in Athena. The exercise has to be done in a calculation center.

Delivery:
This activity is evaluable and corresponds to 30% of the final grade.
It will take place from the third week to the eighth.

Full-or-part-time: 52h
Practical classes: 9h
Laboratory classes: 4h
Self study: 39h

ACTIVITY 3

Description:
A collection of programming exercises (4-5) more complicated than those of activity 2 will be solved. In this case, a class must be implemented that meets the requested specifications.

Specific objectives:
At the end of the proposed exercises the student must be able to:
· Design a class or a set of classes adjusted to the request.
· Implement it and debug it if necessary.

Material:
The documentation will be found in Athena. The exercises must be carried out in the calculation center.

Delivery:
This activity is assessable and corresponds to 30% of the final grade

Full-or-part-time: 36h
Theory classes: 3h
Practical classes: 6h
Self study: 27h

ACTIVITY 4

Description:
The student will design a program for a small project, for example, the change from geocentric Cartesian coordinates to geodetic coordinates and vice versa of a collection of points contained in a text file.

Specific objectives:
At the end of the test you should be able to:

· Do a bibliographic search on the subject in question.
· Design a class with the specified characteristics.
· Implement a program that uses this class.
· Interpretation of results.

Delivery:
In this activity it is evaluable and corresponds to 40% of the final grade.

Full-or-part-time: 40h
Guided activities: 6h
Self study: 34h

GRADING SYSTEM

The final grade is the sum of the following partial grades:

Final mark = 0.30 * mark for activity 2 + 0.30 * mark for activity 3 + 0.40 * mark for activity 4

Attendance and class work will be valued.

BIBLIOGRAPHY

Basic:
- Montejo Ráez, Arturo y Salud María Jiménez. Curso de programación Python. Anaya multimedia, 2017. ISBN 9788441541160.

Complementary:
- Beazlet, David & Brian K. Jones. Python Cookbook. 3rd Edition. O'Reilly, 2013. ISBN 9781449340377.