UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

Course guide
804226 - PROG1VJ - Programming I

Last modified: 13/09/2023

Unit in charge:	Image Processing and Multimedia Technology Centre
Teaching unit:	804 - CITM - Image Processing and Multimedia Technology Centre.
Degree:	BACHELOR'S DEGREE IN VIDEO GAME DESIGN AND DEVELOPMENT (Syllabus 2014). (Compulsory subject).

Academic year: 2023   **ECTS Credits:** 6.0   **Languages:** Catalan, Spanish, English

LECTURER

Coordinating lecturer:	Díaz Iriberri, Jose
Others:	Martín Mínguez, Mónica

DEGREE COMPETENCES TO WHICH THE SUBJECT CONTRIBUTES

Specific:
3. Use programming languages, algorithmic patterns, data structures, visual programming tools, game engines and libraries for the development and prototyping of video games, in any genre and for any platform and mobile device.

Generical:
1. Interpret the basics of the use and programming of computers, operating systems, databases and in general, computer programs with applications for engineering.
2. Interpret and master the basics of discrete mathematics, logic, algorithmics and computational complexity, and their application to the automatic processing of information using computer systems and their application for solving engineering problems.

Transversal:
4. EFFICIENT ORAL AND WRITTEN COMMUNICATION. Communicating verbally and in writing about learning outcomes, thought-building and decision-making. Taking part in debates about issues related to the own field of specialization.
5. EFFECTIVE USE OF INFORMATION RESOURCES: Managing the acquisition, structuring, analysis and display of data and information in the chosen area of specialisation and critically assessing the results obtained.

TEACHING METHODOLOGY

Lectures are divided in 2h sessions. During lectures, the teacher presents the theoretical concepts and explains them by means of examples that are solved in class. Some time is also dedicated to the resolution of exercises by the students with the assistance of the teacher (solve the doubts that may appear). An extensive use of campus virtual (Atenea) is mandatory, since the material of the course (slides, exercises, exams, etc.) and the communication between the students and the teacher are done through this system.
Tasks of the course: There are three types of tasks that are planned during the course:
- Two practical tests/exercises to evaluate specific topics of the course (P).
- A midterm exam that is done in the midterms week 7, specified in the academic calendar (EP).
- The final exam (EF).

LEARNING OBJECTIVES OF THE SUBJECT

- Show understanding and ability to explain the concept of algorithm and to translate simple algorithms expressed in natural language into functions correctly.
- Be able to design and implement small programs with their corresponding test batteries under a given specification.
- Show understanding and ability to explain the concepts of type, value and variable.
- Be able to differentiate between interpretation and compilation.
- Be able to carry out calculations that require the storage of intermediate results in variables, taking into account the correct use of types.
- Be able to implement functions with and without parameters and a return value.
- Be able to use existing libraries in the development of programs.
- Be able to implement programs performing line-by-line input/output actions on text files; programs that use repeat control structures (possibly nested and combined with selection structures): loops with stop condition and with iteration control variables, and programs that handle strings, vectors, buffers and matrices.

STUDY LOAD

Type	Hours	Percentage
Hours medium group	16,0	10.67
Self study	90,0	60.00
Hours large group	24,0	16.00
Guided activities	20,0	13.33

Total learning time: 150 h

CONTENTS

1.Basic concepts and algorithms

Description:
1.1. Definitions: Algorithm and Program.
1.2. Programming languages. Classification. C language.
1.3. Compiling and linking.
1.4. Execution.
1.5. C compiler. Features.
1.6. Examples of games programmed in C.
1.7. Introduction to Visual Studio.

Related activities:
Combine theoretical and practical activities.

Full-or-part-time: 11h
Theory classes: 4h 30m
Self study : 6h 30m

2.Types, operators and expressions

Description:
2.1. Variable Names
2.2. Data types and sizes
2.3. Constants
2.4. Declarations
2.5. Arithmetic operators
2.6. Relation and logical operators
2.7. Casting types
2.8. Increment and decrement operators
2.9. Assigment operators and expressions

Related activities:
Combine theoretical and practical activities.

Full-or-part-time: 10h
Theory classes: 4h
Self study : 6h

3.Control flow

Description:
3.1 Statements and blocks
3.2 If-else
3.3 Else-if
3.4 Switch
3.5 Loops-while and for
3.6 Loops-do-while
3.7 Break and continue

Related activities:
Combine theoretical and practical activities.

Full-or-part-time: 25h
Theory classes: 10h
Self study : 15h

4.Functions and program structure

Description:
4.1. Basics of Functions
4.2. Functions returning non-integers
4.3. External variables
4.4. Scope rules
4.5. Header files
4.6. Static variables
4.7. Register variables
4.8. Block structure
4.9. Initialization
4.10. The C preprocessor
4.11. Standard libraries

Related activities:
Combine theoretical and practical activities.

Full-or-part-time: 30h
Theory classes: 12h
Self study : 18h

Midterm exam

Description:
Midterm Exam

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

5.Arrays

Description:
5.1 One-dimensional arrays
5.2 Multidimensional arrays
5.3 Traversals and searches

Related activities:
Combine theoretical and practical activities.

Full-or-part-time: 30h
Theory classes: 12h
Self study : 18h

6.Structures

Description:
6.1. Basic structures
6.2. Structures and functions
6.3. Arrays of structures
6.4 Typedef
6.5 Unions

Related activities:
Combine theoretical and practical activities.

Full-or-part-time: 25h
Theory classes: 10h
Self study : 15h

7.Files

Description:
7.1. Introduction
7.2. Read files
7.3. Write files

Related activities:
Combine theoretical and practical activities.

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

Final exam

Description:
Examen teórico y práctico

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

GRADING SYSTEM

The mark of the subject (NF) is computed as follows:
NF = 0.2 * EP + 0.4 * P + 0.3 * EF + 0.1*PAA

where EP is the mark of the midterm exam, P are the two tests of practical exercises that will be done throughout the course (note that the mark each one corresponds to the 20% of the final mark of the subject), EF is the mark of the final exam and PAA is Participation and learning attitude.

Students who do not pass the subject through continuous assessment will have the possibility of presenting themselves to the re-assessment test. In this test the qualifications of the exams (EP and EF) will be reevaluated. The resulting final grade of subject can not exceed 5.

EXAMINATION RULES.

Different exercises will be proposed at each class that the students have to solve before the next class individually. The solution must be upload to the Virtual Campus before the deadline.

BIBLIOGRAPHY

Basic:
- Kernighan, B.W.; Ritchie, D.M. El lenguaje de programación C. 2ª ed. México: Prentice Hall, 1991. ISBN 9688802050.

Complementary:
- Joyanes Aguilar, L. Fundamentos de programación: algoritmos y estructura de datos [on line]. 5ª ed. Madrid [etc.]: McGraw-Hill, 2020 [Consultation: 06/05/2022]. Available on: https://www-ingebook-com.recursos.biblioteca.upc.edu/ib/NPcd/IB_BooksVis?cod_primaria=1000187&codigo_libro=10211. ISBN 9781456279448.
- Joyanes , L.; Zahonero, I. Algoritmos y estructuras de datos: una perspectiva en c. Madrid: McGraw-Hill, 2004. ISBN 9788448140779.

RESOURCES

Hyperlink:
- Documentació llenguatge C. https://www.cplusplus.com/reference/