Facultad de Informática
Universidad Complutense

Fundamentals of Programming I

Subject's Program
Course 2020-2021

General Data

Subject: Fundamentals of Programming I

Degrees: Grado en Ingeniería Informática
Grado en Ingeniería del Software
Grado en Ingeniería de Computadores
Doble Grado en Ingeniería Informática y Matemáticas
Doble Grado en Administración y Dirección de Empresas e Ingeniería Informática

Credits: 6 ECTS Classes: First semester

Main Instructors

Group A	Jorge Carmona Rúber
Group B	Ramón González del Campo
Group D	Javier Arroyo Gallardo
Group E	Alberto Núñez Covarrubias
Group F	Sonia Estévez Martín
Group G	Luis Hernández Yáñez
Group I	Luis Hernández Yáñez
Other	Double Degree with Mathematics: Ismael Rodríguez Laguna
	Double Degree with ADE: Mercedes Gómez Albarrán

In laboratory classes, other instructors may assist the main ones.

Summary

The main goal of teaching programming is to make students able to build readable, well-documented, correct and easy-to-maintain programs methodically. The subject *Fundamentals of Programming I* is the first contact with programming learning for undergraduates and presents the concepts and techniques of programming for the first time. By means of a good combination of lectures and laboratory classes, students will assimilate these concepts and techniques from the beginning with progressive complexity, acquiring basic abilities and putting into practice what they have learned. A structured programming approach will be followed and the C++ programming language will be the reference language.

Syllabus

Lesson 1 **Computers and Programming**

Introduction to Computer Science. A little history. Programming as a problem-solving task. The concept of algorithm. Programming languages. Language syntax. Introduction to software engineering. Development environments.

Lesson 2 **Types and Instructions I**

Program elements. Editing, compiling and linking. Errors. Program data. Console input/output. Literal values and variables. Identifiers. Basic data types. Constants. Assignment. Data and memory. Operators and expressions. Predefined functions. Instructions. Code blocks. Selection and iteration. User-defined functions.

Lesson 3 **Types and Instructions II**

More about basic data types. Enumerated types. Type conversions. Input/output with text files. Selection instructions. More about conditions. Iteration instructions: loops with a fixed number of iterations; conditional loops. Sequences. Traversal and search schemes. Simple data arrays.

Lesson 4 **Procedural Abstraction**

Descendent design. General form of a subprogram. Parameters. Parameter types. Pass by value/reference. Arguments. Subprogram declaration. Prototypes. Execution model. The functions return a result.

Lesson 5 **Structured Data Types**

Homogeneous collections. One-dimensional arrays and character strings. Heterogeneous collections. Composition. Variable length lists.

Bibliography

- ✓ *C++: An Introduction to Computing* (2nd Edition)
  J. Adams, S. Leestma, L. Nyhoff. Prentice Hall, 1998.
- ✓ *Programming and Problem Solving with C++, Comprehensive Edition*
  N. Dale, C. Weems. Jones and Bartlett Publishers, 2010.
- ✓ *Programación en C++ para ingenieros*
  F. Xhafa et al. Thomson, 2006.

From the creator of the programming language:

- ✓ *Programming: Principles and Practice Using C++*
  B. Stroustrup. Pearson Education, 2009.

Available online (electronic books):

- ✓ *Programming with C++*
  B. L. Juneja, A. Seth. New Age International, 2009. Link to book in UCM Library.
- ✓ *Beginning Programming with C++ for Dummies*
  S. R. Davis. John Wiley & Sons, 2014. Link to book in UCM Library.

Subject Development

In all lessons there are theoretical concepts, methods and techniques for the students to assimilate, as well as examples and exercises to experiment with and practice. Also, a project will be proposed for each student to develop a working program that satisfies specifications, runs correctly, and is well implemented. The project will be developed outside class. In laboratory classes doubts about the exercises and projects will be resolved and student work will be assessed.

Students will use several development tools, all of them with free access. In the virtual Campus students will find learning material and several tools that will help in the learning process.

In this group we will follow a modern learning approach: *the flipped classroom*. Students will read in advance the set of slides selected for each class, and all the class time will be devoted to solve doubts and work on exercises. There will be also activities (quizzes, tasks) for the students to participate in.

Grading Scheme

Students should work continuously during the academic course:

- ✓ Read slides in advance and consult suggested bibliography
- ✓ Attend classes regularly
- ✓ Go to tutoring sessions to resolve doubts
- ✓ Work on exercises and activities proposed
- ✓ Participate in Virtual Campus (VC) forums and in proposed activities
- ✓ Develop the project on time and send it via the VC
- ✓ Take the final exam

Time spent in classes is not enough to achieve a good understanding of theory and a good practice in techniques. Students should also spend time at home or school studying and practicing.

One final exam must be taken, consisting of the resolution of exercises and practical cases. Also, additional activities will be frequently proposed for students to develop on their own, after which, the solutions are to be sent through the VC.

Exam, project and additional activities all count towards the course grade:

Final exam (January/July) 60%
Project 20%
Activities 20%

In the final exam and the project, students must be graded with at least 5 over 10.

In July, the grade will be calculated in the same manner. If the student didn't participate in the additional activities the corresponding 15% is lost. There will be another deadline for the project, to submit a new version if it was not submitted or not passed in the first place.