Universidad de Alcalá

TEACHING GUIDE

Computer Systems

Degree in
Telecommunication Technologies Engineering (GITT)
Telecommunication Systems Engineering (GIST)
Telematics Engineering (GIT)
Electronic Communications Engineering (GIEC)

Universidad de Alcalá

Academic Year 2023/2024

1st Year - 1st Semester (GITT+GIST+GIT+GIEC)

Approved by the School Board on June 12th, 2023

TEACHING GUIDE

Course Name:	**Computer Systems**
Code:	**350003 (GITT+GIST+GIT+GIEC)**
Degree in:	**Telecommunication Technologies Engineering (GITT)** **Telecommunication Systems Engineering (GIST)** **Telematics Engineering (GIT)** **Electronic Communications Engineering (GIEC)**
Department and area:	**Automática** **Computer Architecture and Technology**
Type:	**Basic (GITT+GIST+GIT+GIEC)**
ECTS Credits:	**6.0**
Year and semester:	**1st Year - 1st Semester (GITT+GIST+GIT+GIEC)**
Teachers:	Mª del Mar Lendínez Chica
Tutoring schedule:	Check the e-learning publishing environment: https://uah.blackboard.com
Language:	English

1. COURSE SUMMARY

Computing Systems is a basic subject imparted in the first term of the first year in several Engineering degrees: Communication Electronics Engineering, Telecommunication Systems Engineering, Engineering on Telecommunication Technologies, and Telematics Engineering.

This subject is aimed to teach how to build computer programs using the basic tools provided by a general-purpose programming language. In order to achieve the goal, the elementary concepts related to general-purpose computers have to be deal with the subject, therefore every constituent element will be described and justified from the point of view of the provided functionality, likewise, the most relevant cases in information coding, starting with the most elementary forms of data and reaching to medium complexity data structures. Similarly, the methods for coding actions and procedures in the algorithmic form will be presented.

The Programming subject, taught in the second term, is based on this subject. Furthermore, Computing Systems provides the knowledge required in several subjects throughout the degree: Algebra Lineal, Electrónica Digital, Diseño Electrónico, Sistemas Electrónicos Digitales, Sistemas Electrónicos Digitales Avanzados, Programación Avanzada, Programación Visual, Sistemas Operativos Some of them are also offered in English.

In summary, the main concepts gathered in this subject are digital computer architecture, information coding, elements of the C Language, standard Input/Output, functions, control statements, arrays, structures, arrays of structures, pointers, and dynamic memory allocation.

2. SKILLS

Basic, Generic and Cross Curricular Skills.

This course contributes to acquire the following generic skills, which are defined in the Section 3 of the Annex to the Orden CIN/352/2009:

en_TR2 - Knowledge of basic subjects and technologies that enables to learn new methods and technologies, as well as to provide versatility that allows adaptation to new situations.

en_TR3 - Aptitude to solve problems with initiative, decision making, creativity, and to communicate and to transmit knowledge, skills and workmanship, comprising the ethical and professional responsibility of the activity of the Technical Engineer of Telecommunication.

en_TR5 - Easy to handle specifications, regulations and mandatory standards.

en_TR8 - Capacity of working in a multidisciplinary and multilingual team and of communicating, both in spoken and written language, knowledge, procedures, results and ideas related to telecommunications and electronics.

en_TRU2 - Oral and written competencies.

en_TRU3 - Ability to manage information.

en_TRU4 - Autonomous learning skills.

en_TRU5 - Team work.

Professional Skills

This course contributes to acquire the following professional skills, which are defined in the Section 5 of the Annex to the Orden CIN/352/2009:

en_CB2 - Basic knowledge about the computers usage and programming, operating systems, databases and computer programs in engineering applications.

Learning Outcomes

After succeeding in this subject the students will be able to:

RA1. Interpret how simple programs work and binary operations.

RA2. Distinguish in a program the basic data types and control statements.

RA3. Identify the different elements use in the structure like functions, arrays and structures.

RA4. Program with pointers and control the application using dynamic memory

RA5. Code high difficulty programs in an autonomous way using the C language, applying directly to every work or scientific environment.

3. CONTENTS

Contents Blocks	Total number of hours
Fundamentals of computers	2 T + 2 P hours
Fundamentals of programming	14 T + 14 P hours
Structured data types	6 T + 6 P hours
Dynamic memory allocation	6 T + 6 P hours

Schedule

Weeks	Contents
01st	Concept of the computer. The digital computer architecture of Von Neumann. Peripherals. The current computers. Programs and languages. Information coding. Operating System.Exercises. Practice: The Integrated Development Environment, IDE.
02nd	Stages developing a program. Introduce the I/O concept, matrix, control statements, and functions. Primitive data types. Derived types. Type synonyms. Literals. Identifiers. Keywords. Commentaries. Exercises. Practice: Intro to the standard Input/Output, edit, compile, execute and debug a simple program.
03rd	Variables. Symbolic constants. Numeric expressions. Operators. Priority and order of evaluation. Type conversions.Exercises. Practice: Operators, binary operators, and simple Control Statements.
04th	Program structure. Inclusion and substitution directives. Declarations and definitions. Sentences: simple and complex. Exercises. Practice: Operators, binary operators, and simple Control Statements.
05th	Functions: declaring, defining, and calling a function. Arguments passed by value and passed by reference. Scope of variables. Storage classes. Numeric data and character strings. Standard input and standard output streams. Formatted output. Formatted input. Input and output of characters. End-of-line and end-of-file characters. Check input data. Exercises. Practice: Operators, binary operators, and simple Control Statements.
06th	Sequential and Repetitive control statements. Exercises. Practice: Control Statements.
07th	Arrays. Numeric unidimensional arrays. Associative arrays. Character strings. Functions for character strings manipulation. Type and size of an array. Multidimensional arrays. Exercises. Practice: Arrays and Control Statements.
08th	Arrays of character strings. Copying arrays. Working with blocks of bytes. Structures. Arrays of structures. Exercises. Practice: Arrays and Control Statements.
09th	Structures. Structures arrays. Exercises. Practice: Structured Datatypes.
10th	Creating pointers. Operators "address of" and "content of address". Operations with pointers. Pointers and arrays. Exercises. Practice: Structured Datatypes. manipulation.
11th	Pointers to arrays of characters. Arrays of pointers. Pointer to pointer. The array of pointers to arrays of characters. Sorting arrays of character strings. Exercises. Practice: Structured Datatypes. manipulation.
12th	Dynamic memory allocation. Functions for dynamic memory allocation. Exercises. Practice: Pointers and Dynamic Arrays.
13th	Dynamic arrays. Pointers to structures. Pointers as function parameters. Exercises. Practice: Pointers and Dynamic Arrays.
14th	Dynamic arrays of pointers. Pointers to pointers dynamic arrays. Practice: Pointers and Dynamic Arrays.

4. TEACHING - LEARNING METHODOLOGIES. FORMATIVE ACTIVITIES.

4.1. Credits Distribution

Number of on-site hours:	58 hours (56 hours on-site +2 exams hours)
Number of hours of student work:	92
Total hours	150

4.2. Methodological strategies, teaching materials and resources

The methodological strategy training is made up of the following types of activities:

Theoretical sessions

Those sessions are lectures where the basic theoretical content is explained. The student has to acquire the programming philosophy. In view of the very practical issue of the subject, theoretical sessions will include the use of many practical examples.

It will be used all media resources existing in the classroom, such as blackboard, overhead projector, and computer, it allows the lecturer to show students the resolution and implementation of programs and it helps to improve the dynamic of the sessions in the theoretical exposition of the contents.

Students will be encouraged to participate in class through dialogue and collective consultation, exercises, in order to promote collaborative learning. Besides, they will be asked to participate actively in a class by solving exercises on the board in order to pool the solutions obtained by the students.

To keep track of the learning and stimulate interest in the programming, the lecturer may request several activities to resolve by students, individually or in the group.

Lab sessions

The purpose of these sessions is to dip students into the world of programming by guiding programs related to professional and employment. For it, there are computers and software required for developing, compiling, executing, and debugging programs.

Each student can develop the programs individually, but it is advised to form groups of two students.

At least five big exercises will be proposed along the course. These activities will include the implementation of programs, of increasing difficulty, related to the contents and examples explained in lectures.

The content of the practical laboratory activities will gradually grow in size and complexity so that students gradually acquire relevant skills to the subject. Initially, the practical activities will be guided by pseudocode that will show the student the way forward the correct execution of the program. As students are acquiring skills, the pseudocode will be eliminated as a guide so will be given a degree of freedom to perform the encoding of the programs, according to their criterion. The teacher will become a mentoring role, which will consist in guiding the student in performing the work.

During the learning process, students can make use of both, books and information searches on the

Internet.

Tutoring

Tutoring schedules are established to facilitate the support of those students who require it, individually or in groups. Also enables the lecturer to monitor the assimilation of the concepts. This will enable the adaptation of classes related to greater learning.

To support and complement the above activities, students have a learning platform where they can access information and resources of the course such as:

- Documentation developed by the lectures
- Practical documents
- Previous exams files
- Lecturer staff and tutorial schedule
- Updated information about the course such as forums and news
- Schedule of the exams and grades

5. ASSESSMENT: procedures, evaluation and grading criteria

Preferably, students will be offered a continuous assessment model that has characteristics of formative assessment in a way that serves as feedback in the teaching-learning process.

5.1. PROCEDURES

The evaluation must be inspired by the criteria of continuous evaluation (Regulations for the Regulation of Teaching Learning Processes, NRPEA, art 3). However, in compliance with the regulations of the University of Alcalá, an alternative process of final evaluation is made available to the student in accordance with the Regulations for the Evaluation of Apprenticeships (approved by the Governing Council on March 24, 2011 and modified in the Board of Directors). Government of May 5, 2016) as indicated in Article 10, students will have a period of fifteen days from the start of the course to request in writing to the Director of the Polytechnic School their intention to take the non-continuous evaluation model adducing the reasons that they deem convenient. The evaluation of the learning process of all students who do not apply for it or are denied it will be done, by default, according to the continuous assessment model. The student has two calls to pass the subject, one ordinary and one extraordinary.

Ordinary call

Several tests are going to be performed during the term in the continuous assessment. The student, who does the last test will be graded, if a student doesn't perform the last test will be graded as “No submission”.

The single assessment model will be carried through by an only exam, it is going to coincide in date, but not in format and content with the last test in the continuous assessment. Both would be in written mode or a BlackBoard platform test.

Extraordinary call

The students that didn't pass the subject in the normal assessment have another chance to pass the course, using the extra assessment. This model consists of a single exam that gathers questions related to the whole topics of the subject.

5.2. EVALUATION

EVALUATION CRITERIA

The objective of this process is to check the acquisition of the subject competencies. For that are defined the following criteria:

CE1. The student has acquired the programming technical concepts explained during the course and applies them in the right way.

CE2. The student can solve problems correctly and in a conceptual way, dealing with the theoretical topics explained in class. These problems may differ from the exercises solved in the practical sessions.

CE3. The student can analyze and understand problems or technological needs, business, industrial, scientific, and in general from every working area, providing a solution through efficient algorithms and programming.

CE4. The student proves the control on the programming philosophy using correctly the data structures, control statements, pointers, and dynamic memory.

CE5. The student shows a capacity to understand, code, and modify considerate difficulty programs, these programs are built with several functions and files.

GRADING TOOLS

In this section are shown the assessment tools applied to the Evaluation rules. Every test is going to be performed using the Moodle learning platform, hold in Web, and these tests are made up of multi-choice questions and short development.

1. Intermediate assessment test (PEI): It consists of a test, that checks basic knowledge about the subject and groups questions of the half list of topics. The PEI is going to take place in the laboratory and consists of several multi-choice questions along with a programming activity.

2. Laboratory test (PLn): There will be at most four tests where every question is based on activities performed in the practical lab sessions.

3. Final Assessment (PEF): It consists of a multi-choice exam and questions to develop. The PEF will take place on the day proposed by the Escuela Politécnica Superior in the labs of the Automatic Department or in the teaching rooms.

GRADING CRITERIA

This section quantifies the rules of assessment to pass the course.

Regular assessment. Continual evaluation.

In the regular assessment – continual evaluation the relation between the rules, instruments, and mark is gathered in the following table:

Skill		Learning Outcomes	Evaluation criteria	Grading Tool	Contribution to the final mark
TRU4	CB2, TR2, TR3, TR8, TRU2	RA1, RA2, RA3	CE1-CE5	PEI	40%
	TR3, TR5, TR8, TRU2, TRU3, TRU5	RA1-RA5	CE1-CE5	PLn	20%
	CB2, TRU2, TRU3, TRU5, TR2, TR3, TR5, TR8	RA1-RA5	CE1-CE5	PEF	40% (100%)

If the mark obtained in the PEF led to a better mark than the grade obtained in the continuous assessment, the final mark assigned will be 100% of the PEF, otherwise, the weight of this assessment

is going to be 40% of the final mark. Just in case not attending the PEF, the final mark will be No Presentado, (NP).

Attendance to the laboratory: Attendance to 80% of laboratory sessions is mandatory. If a student has an unjustified absence rate exceeding 3 laboratory sessions, the student will get a grade of "No submission" in the regular assessment. During the practical sessions, the students are going to develop the programs proposed in the documentation, they have to pay attention to the errors and fix them to get the correct target. Laboratory work is recognized through examinations, which are all based on the contents of the practices.

The fact to pass just one of the three continuous tests does not mean passing the course.

Regular assessment, Final assessment

Skill	Learning Outcomes	Evaluation criteria	Grading Tool	Contribution to the final mark
CB2, TRU2, TRU3, TRU4, TRU5, TR2, TR3, TR5, TR8	RA1-RA5	CE1-CE5	PEF	100%

Extraordinary call

Skill	Learning Outcomes	Evaluation criteria	Grading Tool	Contribution to the final mark
CB2, TRU2, TRU3, TRU4, TRU5, TR2, TR3, TR5, TR8	RA1-RA5	CE1-CE5	PEF	100%

This exam will take place the day proposed by the Escuela Politécnica Superior for extraordinary calls in the labs of the Automatic Department or in the teaching rooms and it will consist of a multi-choice exam and programming questions to develop.

6. BIBLIOGRAPHY

6.1. Basic Bibliography

Basic bibliography in English:

- C How to Program, Deitel & Associates Inc. H.M.Deitel & P.J.Deitel
- The C Programming Language. Prentice-Hall. B. Kernighan and D. M. Ritchie.

Basic bibliography in Spanish:

- Curso de programación con C/C++. Ed. RA-MA. Fco. Javier Ceballos.

Disclosure Note

During the evaluation tests, the guidelines set out in the Regulations establishing the Rules of Coexistence of the University of Alcalá must be followed, as well as the possible implications of the irregularities committed during said tests, including the consequences for committing academic fraud according to the Regulation of Disciplinary Regime of the Students of the University of Alcalá.