ETSI Informática
University of
Malaga

Foundations of Computer Science

Course Guide 2022-2023

http://www.informatica.uma.es

Foreword

We welcome you to study at the School of Computer Science at the University of Malaga (ETSI Informática – UMA). In this course guide, which is first and foremost designed to help exchange students to plan their studies during the exchange period, you will find descriptions of all the courses taught in English at our School during the academic year 2022-2023. Further inquiries will be handled by your contact teachers and the International Coordinator of the School:

Julián Ramos Cózar
secretario@informatica.uma.es

Also, the International Office is there to help you:

Laura Mena and M.ª Magdalena de la Chica
incoming@uma.es
http://www.uma.es/relaciones-internacionales/

We hope you will enjoy your stay at University of Malaga.

http://www.uma.es

For more information and the official programs* visit our website:
http://www.informatica.uma.es

*** Disclaimer:** This document is a succinct translation of the original programs in Spanish. In no case the contents of this translation may be interpreted as a replacement or modification of the conditions specified in the original documents.

Course Syllabi

CODE 101

NAME Calculus for computing

Credits 6 ECTS

Period Fall Semester

Course Specifications
The training activities will be based on theory, problems, and practical classes, and on the student's personal work with the exercises and other material provided by the lecturer.
The active work of the student will be encouraged to solve the exercises proposed and practical exercises will be done with mathematical software.

Objectives and contents
In this subject, the fundamentals for solving mathematical problems involving knowledge of differential and integral calculus are developed.

Contents
1. Elementary functions. Solving equations and systems. Complex numbers.
2. Differential calculus.
3. Integral calculus.
4. Sequences and series.

Assessment
Continuous assessment, partial tests for each subject, with which the student can pass the course. A final exam is also possible as an alternative to pass this subject.

Lecturer
Dr. Sergio Ortega Acosta
sergio.ortega@uma.es
Room 2.2.16

CODE 104

NAME Fundamentals of Programming

Credits 6 ECTS

Period Fall Semester

Course Specifications
Learning how to program computers. Most of this course hard work is to be performed in front of a computer. The theoretical basis and concepts are not complex and are previously discussed in classes.

Objectives and contents
The main objective is acquiring correct programming skills. A general programming language is used. References are made to other languages and techniques. The main focus is on the acquisition of algorithmic construction skills and code organization. Understanding the underlying concepts of computational sciences and a light approach to its formal basis will provide the necessary background.

Contents:
1. Introduction to programming.
2. Introduction to a programming language: C/C++
3. Procedural abstraction.
4. Structured data types

Assessment
Attending lectures is compulsory. Apart from the final unavoidable exam, there will be, at least, two midterm exams. Every practice, homework exercise, class activity, and forum participation will be part of a continuous evaluation.

Lecturer
Dr. Juan Falgueras Cano
juanfc@uma.es
Room 3.2.32

Course Syllabi

CODE 106

NAME Algebraic Structures for Computing

Credits 6 ECTS

Period Spring Semester

Course Specifications
Lectures in the lecture room. Problem solving with and without computers support in the lecture room. Individual work is required.

Objectives and contents
The course on Algebraic Structures for Computing will set the foundations for the students to be trained on specific formal languages that will be useful for them in future courses such as ´Algorithms Design`, ´Distributed Systems and Networks`, ´Automata Theory and Formal Languages` and many others.

Contents:

1. Preliminary Concepts
2. Lattices and Boolean Algebra
3. Groups, Rings and Fields
4. Linear Equations Systems
5. Vector Space
6. Linear Functions
7. Diagonalization
8. Inner Product Space

Assessment
The attitude and work done by the student during the classes will be taken into consideration. During the practical sessions of the course, the active participation of the student will be encouraged in order to solve the proposed exercises.

May/June Exam - Continuous evaluation

(35%) Proposed activities and exercises to be handed-in with or without computer tools, work in the on-site class, questionnaires, etc. The number of these activities will depend on the organization of the evolution of the course as well as on the real number of students enrolled in it.

(65%) Partial Exams.

In order to take into consideration, the above marks (the 35% and the 65%), the student must obtain at least 3.5 out of 10 in each of the parts described above. Students who do not pass the course following this procedure could do the final exam on the date the School schedules it.

Only the marks of the students who pass the course by continuous evaluation or those who do the final exam will be reflected in their official marks record.

Other exams

For the second and extraordinary exams, there will be a unique exam on the contents of the whole course on the date scheduled by the School.

Lecturer
Dra. Mª Carmen Fernández Gago
mcgago@uma.es
Room 2.2.22

Course Syllabi

CODE 201

NAME Analysis and Design of Algorithms

Credits 6 ECTS

Period Fall Semester

Course Specifications
Flipped-classroom model. In-classroom activities: discussion sessions, problem solving sessions, laboratory work; Individual work: online lectures, problem assignments.

Objectives and contents
The objectives of the course are (1) knowing and grasping the main techniques for algorithm design (i.e., being able to apply these for solving specific problems and being able to reason about their applicability and suitability) and (2) attaining adequate knowledge about algorithmic complexity (i.e., analyze the algorithms built, reason about their efficiency and perform comparisons among algorithms).
The course is organized in two major thematic parts, one dealing with algorithm analysis (complexity, formal specification and verification) and another one tackling algorithm design (comprising techniques such as divide-and-conquer, dynamic programming, greedy algorithms, backtracking and branch-and-bound).

Assessment
A continuous assessment methodology is used: three partial exams and several lab tests will be conducted during the semester. The final mark will be based on their outcome plus a bonus obtained by actively participating in class (e.g., delivering proposed assignments, solving problems in class, etc.). A final exam will be available for students who could not obtain enough points during the semester.

Lecturer
Dr. Carlos Cotta Porras
ccottap@lcc.uma.es
Room 3.2.49

Course Syllabi

CODE 202

NAME Databases

Credits 6 ECTS

Period Fall Semester

Course Specifications
Lectures in the classroom will introduce the foundations of database systems. Practical activities will allow the students to learn how to handle a commercial database manager system. The student will put in practice the basic topics and tools concerning database systems. This course is mainly developed in the labs, directly interacting with professional database software.

Objectives and contents
The objectives are to learn the concepts of the relational model and to use these elements via a commercial database manager. Contents include the Entity-Relationship and Relational models, the data definition language and data manipulation language of SQL and an introduction to the database architecture.

Assessment
There will be 2 main activities to be developed and evaluated in the semester (40% of the global score): database design (project activity) and database implementation by using the Oracle software (lab activity). A final exam regarding SQL language (lab activity) and a questionnaire complete the evaluation. The students have to engage in all the activities, participate in every evaluation milestone, and show a minimum level in all of them.

Lecturers
Dr. Manuel Enciso García-Oliveros
enciso@lcc.uma.es
Room 3.2.30

Dr. David Bueno Vallejo
david.bueno@uma.es
Room 3.2.27

CODE 203

NAME Computer Organization

Credits 6 ECTS

Period Fall Semester

Course Specifications
Lectures in the classroom, individual work, questionnaires on-line by virtual campus and laboratory work

Objectives and contents
The objective of the subject is to understand the internal organization of a computer (from a hardware point of view). The contents are divided in three units. The first one is composed of two chapters**:** Measuring and understanding performance (~2 weeks) and enhancing performance with Pipelining (~4 weeks). This unit has a laboratory exercise: Simulation of a pipelined processor (DLX). The second unit is composed by a unique chapter: Memory hierarchy (~4 weeks) and we use a cache simulator in the laboratory. Finally, the last unit is composed by one chapter: Input/output and peripherals (~3 weeks) with a laboratory work: Example of an Input/output system: ARM processor on Raspberry Pi

Assessment
There will be four partial exams (~80% of the final grade) and other exercises (~20% of the final grade): list of exercises, problems, workshops, and practical assignments in the laboratory. A final exam is mandatory if the partial exams are failed.

Lecturer
Dr. Julio Villalba Moreno
jvillalba@uma.es
Room 2.2.38

Course Syllabi

CODE 204

NAME Data Structures

Credits 6 ECTS

Period Fall Semester

Course Specifications

Lectures and laboratory work.

Objectives and contents

Data structures are nowadays used to organize huge amounts of data so that algorithms can process them efficiently. This course introduces most important data structures used in Computer Science. Previous familiarity with programming using an object oriented language like Java is presumed. Different implementations of the structures using such programming language and a functional one will be described and corresponding performances will be analyzed and compared. Applications of data structures to solve different problems will also be presented.
Contents

- Functional Programming.
- Linear data structures: stacks, queues and lists.
- Trees: search trees, heaps and balanced trees.
- Hash tables: hash function and, collision resolution techniques.
- Graphs: directed and undirected graphs, depth and breadth first traversals, topological sorting.

Assessment

Continuous assessment during the lectures and laboratory sessions. Written and on computer examinations.

Lecturer

Dr. José E. Gallardo
pepeg@lcc.uma.es
Room 3.0.5

CODE 205

NAME Automata Theory and Formal Languages

Credits 6 ECTS

Period Fall Semester

Course Specifications

The course is mainly based in lectures in which student participation is encouraged through the presentation of exercises, individual work, and group work. Extra activities include writing essays and solving proposed exercises.

Objectives and contents

The course gives a basic introduction to the classic and contemporary theory of formal languages and automata theory.
Contents include the following topics:
Mathematical preliminaries, Alphabets and Grammars, Regular Expressions, Finite automata, Regularity conditions, Context Free Languages, Introduction to computation, The Turing Machine, Recursive functions, The "While" language, Equivalence theorem, Universality, Formal limits of computation.

Assessment

The course contents will be evaluated in 4 blocks. A minimum of 5 is needed in each of the 4 blocks to pass the course, and for achieving this, the students have one opportunity during the continuous evaluation and a second and third ones in the two ordinary final exams calls. The final grade will be obtained by averaging the best grades obtained from each of the 4 blocks plus some extra points from proposed activities.

Lecturer

Dr. Leonardo Franco
lfranco@lcc.uma.es
Room 3.2.29

Course Syllabi

CODE 920

NAME Computational Intelligence

Credits 4.5 ECTS

Period Fall Semester

Course Specifications

The course is based on lectures in which the main ideas of the course are given as specific details of the problems and algorithms to be discussed and applied should be taken from reading the suggested related scientific publications. For each of the topics of the course, a computer based practice should be carried out together with a report of the work done and the results obtained.

Objectives and contents

The objective of the course is that the students acquire the knowledge and abilities needed in order to choose and apply computational intelligence algorithms for solving real problems in bioinformatics.

Contents:

1. Clustering algorithms application to DNA microarray data.
2. Classification algorithms application for prediction of disease evolution.
3. Evolutive algorithms application to feature selection.
4. Probabilistic algorithms application to sequence detection and alignment.

Assessment

Course assessment is based on class participation, on the elaboration of reports and on a final exam.

Lecturers

Dr. Leonardo Franco
lfranco@lcc.uma.es
Room 3.0.29

Dra. Rafaela Benítez
benitez@lcc.uma.es
Room 3.2.21

CODE 932

NAME Biomedical Computational Modeling

Credits 4.5 ECTS

Period Fall Semester

Course Specifications

Lectures to explain theoretical concepts, individual laboratory work with programming practices, work in groups with specific applications and seminars. The course follows a practical approach by using the MATLAB and MODELICA programming languages.

Objectives and contents

The aim of this course is to provide an introduction to the computational modeling of biomedical systems, presenting the underlying principles of mathematical modeling methodology together with its implementation using programming languages. This course is directed at students across a wide range of engineering background and is designed to appeal to biomedical engineers and to others studying physical and engineering sciences, and biological and life sciences by following an interactive computational approach. Outline:

1. Introduction to System Modeling
2. Mathematical Modeling of Biomedical Systems
3. Computational Modeling Tools. MATLAB and MODELICA
4. Identification of Biological Systems
5. Modeling of Physiological Control Systems
6. Dynamic Modeling of Biological Systems

Assessment

Continuous assessments, collaborative work in groups, questionnaires, a final laboratory exam.

Lecturer

Dr. Javier Fernandez de Canete
ffernandezr@uma.es
Room 2.2.29

Course Syllabi

CODE 109

NAME Object Oriented Programming

Credits 6 ECTS

Period Spring Semester

Course Specifications
Lectures where the theoretical concepts are described. Individual and laboratory work where the students put in practice these theoretical concepts using the Java programming language.

Objectives and contents
The course introduces students to object oriented design and programming using Java.

Course Outline:
1. Object Oriented Programming Fundamentals: classes, objects, methods, messages, composition, inheritance, data polymorphism, dynamic binding.
2. Introduction to Java.
3. Exception management.
4. Basic predefined classes in java.lang, java.util and java.io.
5. Collections.
6. Graphical User Interfaces.

Assessment
Attending the lectures and the laboratory sessions, short questions to be answered during lectures, short programs to be developed during laboratory sessions, partial and final exams in the laboratory.

Lecturer
Dr. José Mª Álvarez Palomo
jmalvarez@uma.es
Room 3.2.6

CODE 110

NAME Computer Technology

Credits 6 ECTS

Period Spring Semester

Course Specifications
The asynchronous part is based on studying material based on slides, videos and a book and working on some exercises at home that are uploaded into the system. In the synchronous class, we are mainly working in varying teams using English language on exercises and laboratory assignments presenting elaborations followed by a discussion. For the exchange of ideas, we use tasks, workshops, fora, etc. that are available in the CV system.

Objectives and contents
Go into the wonderful world of the computer processor. The student designs processors based on Boolean logic and observes its functioning based on code we feed it. We get a feeling for the language the processor understands based on creating assembler code.

Assessment
The English of the student is promoted due to interacting in the teamwork and discussions. The assessment of the course is done in a continuous way by providing 6-10 small tests evaluating each part after it is finished. The handed in work of assignments during the course are directly used in the tests.

Lecturer
Dr. Eligius M.T. Hendrix
eligius@uma.es
https://sites.google.com/site/eligiushendrix/

Course Syllabi

CODE 206

NAME Introduction to Software Engineering

Credits 6 ECTS

Period Spring Semester

Course Specifications
The course is mainly based on lectures in which student participation is encouraged through classroom debates and collaborative development of exercises. Students will work on assignments both individually and in groups. Laboratory work will be used to teach selected software engineering tools and to develop the ability to apply the theoretical knowledge in practical scenarios. Extra activities include peer reviews, writing essays and solving proposed exercises.

Objectives and contents
This is an introductory course in Software Engineering. As such, and given the wide domain of the subject in terms of techniques and tools, the course is necessarily dense in content. We offer a broad view of the problems that are related to software development, the concepts, processes, techniques, tools and standards that conform what is called the Software Engineering Body of Knowledge. Our goal is to provide students with a good understanding of this central aspect of software development that will allow them to take more advanced courses on specific subfields.

Assessment
Continuous assessment of the student participation in lectures, debates and laboratory sessions throughout the course. Evaluation of student assignments. Written and practical (on computer) examinations.

Lecturer
Dr. Antonio Maña Gómez
amg@lcc.uma.es
Room 3.2.16

CODE 207

NAME Systems Programming and Concurrency

Credits 6 ECTS

Period Spring Semester

Course Specifications
Lectures where theoretical concepts, problems and algorithms are explained and laboratory work where every student put in practice these concepts resolving well-defined problems. The course follows a practical approach using the Java and C programming languages.

Objectives and contents
The main objective is to learn parallel programming in contrast to traditional programming models. Classical problems are shown as well as their solutions using several approaches. A second objective is to learn the C programming language from a low level point of view (system programming). Contents include:

1. Systems programming: an introduction.
2. Low level programming in C.
3. Concurrency: main concepts.
4. Concurrency: communication and synchronization.
5. Event driven programming.

Assessment
Practical exams focused on low level C capabilities, threads in Java and Concurrency using shared memory. In addition, participation in class and a final homework on Swing and background threads will be part of the final mark.

Lecturer
Dr. Sergio Gálvez Rojas
galvez@uma.es
Room 3.2.28

Course Syllabi

CODE 208

NAME Networks and Distributed Systems

Credits 6 ECTS

Period Spring Semester

Course Specifications
Lectures and problem solving sessions; individual and laboratory work.

Objectives and contents
This is the first course about *Communication Networks* with focus on communication protocols covering from physical level thru the application level.

Contents include the following topics:
Internet layered network structure. Basic protocol functions such as addressing, multiplexing, routing, forwarding, flow control, re-transmission error recovery schemes, and congestion control. Overview of link, network and transport layer protocol standards, following a bottom-up approach. Introduction to wireless and mobile networks. This course will also give hands-on experience in network programming using the socket API in C and Java programming languages.

Assessment
The acquisition of concepts is evaluated considering: class attendance and participation; midterm and final exams (70% aprox.). The practical part of this course counts 30% of the final grade.

Lecturer
Dra. Lidia Fuentes
lff@lcc.uma.es
Room 3.2.8

CODE 209 (305)

NAME Intelligent Systems

Credits 6 ECTS

Period Spring Semester (Fall Semester)

Course Specifications
Lectures, written exercises, individual work, programming laboratory practices.

Objectives and contents
This is a first course in Artificial Intelligence (AI). AI is 'the art of creating machines that perform functions that require intelligence when performed by people' (Kurzweil, 1990). We offer a broad view of the problems that AI can solve. The aim is to provide a general knowledge of this branch of Computer Science, so that you are able to take more advanced courses on specific subfields. Course contents include: search, games, logic, planning, artificial neural networks and decision problems.

Assessment
There are two written exams: a midterm exam and a final exam. The grade of the midterm exam is used to improve the grade of the final exam, in case that the grade of the final exam is lower.
Also, there are mandatory programming exercises, to be coded in Java. Therefore, programming skill is a prerequisite. Active class participation and other optional activities give extra marks.

Lecturer
Dr. Ezequiel López-Rubio
ezeqlr@lcc.uma.es
Room: 3.2.42

Course Syllabi

CODE 210

NAME Operating Systems

Credits 6 ECTS

Period Spring Semester

Course Specifications
The course includes lectures, exercises, and laboratory work (practical demonstrations and individual work on programming).
The course requires **strong** knowledge of the structure of the computer (knowledge of previous subjects like Computer Technology and Computer Structure is compulsory as the concepts will appear frequently during description of the implementation of the OS).

Objectives and contents
The subject focuses on the Operating System as an interface between the developer and the computer hardware in a general purpose system. Any kind of usage of the computer is done through the Operating Systems and never directly:

- High Level Languages access hardware resources through the services on the OS even when the developer uses libraries of the language.
- Programming languages offer access to abstractions which are actually implemented by the OS like threads, processes, files, network sockets, memory allocation, etc.

Contents:

- Processes and threads
- Process scheduling
- Memory management
- Input / Output

Assessment
For the standard student, continuous evaluation of the progress is compulsory. If the student misses an intermediate exam or does not hand in laboratory work before the indicated deadline, there will be NO FINAL exam to be taken as a replacement (with exception of properly documented causes like medical reasons, severe illness or decease of close relatives, etc.)
The assessment involves two parts. To pass the subject, it is required to pass both parts with a minimum score of 5.

- Evaluation of knowledge of theory and solving of problems during the course. This evaluation will be done through two kinds of events:
  - Intermediate exams.
  - Exercises interleaved with theory classes. Attendance to classes is compulsory as missing exercise will count as 0 in the continuous evaluation.
- Evaluation of the laboratory work. Evaluation will include two kind of events:
  - Handing in the output of the laboratories through the virtual campus.
  - Exercises interleaved with laboratory sessions. Attendance to sessions is compulsory as part of the work consist in group exercises during classes.

Lecturer
Dr. Ricardo Quislant del Barrio
quislant@uma.es
Room 2.2.25

Program Schedule

Full year offer (99 ECTS)

Fall Semester	Spring Semester
Calculus for Informatics (6)	Algebraic Structures for Computing (6)
Fundamentals of Programming (6)	Object Oriented Programming (6)
Analysis and Design of Algorithms (6)	Computer Technology (6)
Databases (6)	Introduction to Software Engineering (6)
Computer Organization (6)	Systems Program and Concurrency (6)
Data Structures (6)	Networks and Distributed Systems (6)
Automata Theory and Formal Lang. (6)	Intelligent Systems (6)
Computational Intelligence (4.5)	Operating Systems (6)
Biomedical Computational Modeling (4.5)	

http://www.informatica.uma.es