ETSI Informática
University of Málaga

Foundations of Computer Science

Course Guide
2024-2025

http://www.informatica.uma.es

Welcome

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 incoming 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 **2024-2025**. Further inquiries will be handled by the 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:

Patricia Prados y Diana Gardumi
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.

Academic Calendar

ESCUELA TÈCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA

CALENDARIO OFICIAL GRADOS 24|25

1er semestre/1st term: De/From 10/09/2024 a/to 20/12/2024. 2º semestre/2nd term: De/From 14/02/2025 a/to 06/06/2025. Exámenes convocatoria extraordinaria/Extraordinary exams (instructional days): De/From 16/10/2024 a/to 31/10/2024. Exam. 1er sem. 1ª Ord./1st Term 1st Exams: De/From 07/01/2025 a/to 27/01/2025. Exam. 1er sem. 2ª Ord./1st Term 2nd Exams: De/From 04/02/2025 a/to 13/02/2025. Exam. 2º sem. 1ª Ord./2nd Term 1st Exams: De/From 09/06/2025 a/to 27/06/2025. Exam. 2º sem. 2ª Ord./2nd Term 2nd Exams: De/From 04/07/2025 a/to 15/07/2025

- Lectivo/Instructional Day
- Inicio y fin de semestre/Start & end of term
- Exámenes ordinarios/Ordinary exams
- No lectivo/No lectures
- Exámenes extraordinarios/Extraordinary exams
- Actividades culturales/Cultural Activities (Hackers week and days)
- Festivo/Public Holiday

UNIVERSIDAD DE MÁLAGA | uma.es

E.T.S. INGENIERÍA INFORMÁTICA
UNIVERSIDAD DE MÁLAGA

Course Syllabi

Fall Semester (70.5 ECTS)

Code	Year	Name (ECTS)
101	1	Electronic Foundations (6)
102	1	Physics Foundations of Computer Science (6)
103	1	Introduction to Programming (6)
104	1	Discrete Mathematics (6)
201	2	Analysis and Design of Algorithms (6)
202	2	Databases (6)
203	2	Computer Organization (6)
204	2	Data Structures (6)
205	2	Statistical Methods (6)
305	3	Intelligent Systems (6)
920	3/4	Computational Intelligence (4.5)

Spring Semester (54 ECTS)

Code	Year	Name (ECTS)
106	1	Calculus for Informatics (6)
107	1	Algebraic Structures (6)
108	1	Introduction to Software Engineering (6)
109	1	Advanced Programming I (6)
110	1	Computer Technology (6)
206	2	Foundations of Artificial Intelligence (6)
207	2	Advanced Programming II (6)
208	2	Networks and Services (6)
209	2	Operating Systems (6)
210	2	Automata Theory and Formal Languages (6)

Course Syllabi

CODE 101

NAME Electronic Foundations

Credits 6 ECTS

Period Fall Semester

Course Specifications

This subject aims to provide a comprehensive overview of the different aspects of digital electronics, starting from a level of physical devices and reaching a description of logical devices. The course is composed of theory lessons, problem solving classes and laboratory practices.

Objectives and contents

The main objective of the subject is to provide the students with a first approach to the electronics fundamentals of informatics, helping them to understand the basic elements behind common computer devices (such as processors, memories, etc.) and their operation.

Contents

1. Electronic concepts
2. Electronic devices
3. Switching electronics. Logic families
4. Introduction to digital systems
5. Gate-based combinational analysis and design
6. Combinational logic blocks
7. Sequential logic

Assessment

Regarding the assessment of the subject, it is performed taking into account laboratory practices (30%) and a final exam (70%). Besides, there are two mid-course exams. Passing the mid-course exams implies removing their content from the final exam.

Lecturer

Dr. Andrés Trujillo León
atrujilloleon@uma.es
Room 1513-D (Escuela de Ingenierías)

Course Syllabi

CODE 102

NAME Physical Foundations of Computer Science

Credits 6 ECTS

Period Fall Semester

Course Specifications

The course aims to provide computer engineering students with the knowledge that will enable them to understand the basic physical aspects of electronic devices, especially those related to computer technologies. The methodology is a combination of lectures, problem solving classes and practical laboratory classes. The course also aims to collaborate in inculcating the procedures and rigor of the scientific method within the framework of the development of their future work as engineers.

Objectives and contents

This course develops the fundamentals of electromagnetic theory and quantum and solid-state physics, with emphasis on semiconductors and computer applications.

Contents

1. Electric Field.
2. Magnetic Field.
3. Electromagnetic waves.
4. Fundamentals of Quantum Physics.
5. Introduction to Solid-State Physics.
6. Semiconductor Physics.

Assessment

Continuous assessment with two partial exams and a laboratory exam, with which the student can pass the course. A final exam will be available for students who could not obtain enough points during the semester.

Lecturer

Dr. Emilio Ruiz Reina
eruizr@uma.es
Room 2.3.5A-B

Course Syllabi

CODE 103

NAME Introduction to Programming

Credits 6 ECTS

Period Fall Semester

Course Specifications
Learning computer programming. Most of the hard work in this course will be done in front of a computer. The theoretical background and concepts are not complex and are previously discussed beforehand in the classes.

Objectives and contents
The main objective is to acquire correct programming skills. A general programming language is used. References are made to other languages and techniques. The focus is on acquiring skills in algorithmic construction and code organisation. An understanding of the underlying concepts of computer science and a light approach to its formal basis will provide the necessary background.

Contents:
1. Computer science and programming
2. Introduction to a programming language
3. Procedural abstraction
4. Structured data types

Assessment
Class attendance is compulsory. Apart from the unavoidable final exam, there will be at least two mid-term exams. Each practical, homework exercise, class activity and participation in forums and mid-term exams will be part of a continuous assessment.

Lecturer
Dr. Juan Falgueras Cano
juanfc+bil@uma.es
Room 3.2.19

Course Syllabi

CODE 104

NAME Discrete Mathematics

Credits 6 ECTS

Period Fall 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 Discrete Mathematics contents of this course provide a solid foundation in mathematics and analytical reasoning, while providing the mathematical tools necessary for analysis, optimization and decision making in software design and development. Their understanding allows evaluating the efficiency of algorithms, optimizing computer systems and developing intelligent software solutions based on data analysis.

Contents

1. Preliminary concepts and Number Theory.
2. Sets, functions, counting and recurrence relations.
3. Binary relations and graphs.
4. Classical Logic.

Assessment

In general, in the assessment of students, their attitude and work during the teaching sessions will be assessed. In the practical sessions of the course, the student's active work will be encouraged to solve the exercises proposed in each of the topics. In all sessions, the student will be able to use the computer software recommended in the course.

First Call

The course will follow a continuous assessment that will consist of the delivery of activities and exercises solved with or without computer support, classroom work, questionnaires distributed throughout the course, partial exams... The number of these activities will merely depend on the organization and evolution of the course and may be conditioned by the coordination criteria established by the School and the evolution of the actual size of the groups.

The contents of the course are divided into 4 topics. The final grade for the course is a weighted average with the following weights: 25% for topic 1, 30% for topic 2, 25% for topic 3 and 20% for topic 4.

The weight of the assessment of the course during the teaching period will be 100%. On the date scheduled by the School of Informatics for the first ordinary call, exams will be carried out, as another opportunity to pass the topics failed during the teaching period. The grade obtained in each topic in the recovery exam will replace the corresponding grade for that topic, obtained in the teaching period, and the weighted average grade will be recalculated and that will determine the final grade.
A student's grade will only be transferred to the record when he or she passes the course during the teaching period or appears in the first ordinary session.

Second Ordinary Call
On the date scheduled by the school, the exam of all the content of the course will have to be taken and will account for 100% of the grade. The grade obtained will be transferred to the official records of the call with the corresponding grade.

Extraordinary Call
On the date scheduled by the center, the exam of all the content of the course will have to be taken and will account for 100% of the grade. The grade obtained will be transferred to the official records of the call with the corresponding grade.

Lecturer
Dra. Mª Carmen Fernández Gago
mcgago@uma.es
Room 2.2.21

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 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 thematic parts, a short one dealing with algorithm analysis (introductory topics, computational complexity) and another one –which will comprise most of the course– tackling algorithm design (including techniques such as divide-and-conquer, dynamic programming, greedy algorithms, backtracking and branch-and-bound).

The overall focus of the course is on applied matters and practical problem solving with algorithms. Programming knowledgeability is assumed.

Assessment

A continuous assessment methodology is used: three partial tests and several lab projects 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.).

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

Course Syllabi

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 no final exam in the first ordinary call, where the continuous assessment method is applied. Four partial exams (P) will be taken (P1, P2, P3, P4) such as average_note = 0.1*P1+0.3*P2+0.3*P3+0.3*P4. No minimum mark per control is required except in the last control (3 points minimum). Students with a mean note greater than or equal to 5 will add up to 2 additional points from the activities proposed by the teacher during the course, saturating the resulting value at 10.

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
This course provides an in-depth exploration of data structures, which are essential for organizing large datasets to enable efficient algorithmic processing. Aimed at students with prior experience in imperative programming using languages such as C, and object-oriented programming with Java, the course offers a comprehensive introduction to the most significant data structures in the field of Computer Science.

Participants will gain insights into various implementations of these structures using previously mentioned programming languages. The course will analyse and compare the performance of these implementations, providing a clear understanding of their efficiencies. Additionally, the practical application of these data structures in solving diverse problems will be demonstrated, highlighting their importance in computational tasks.

Contents

1. C Programming Language Review. An examination of dynamic memory management and the implementation of array-based and linked data structures in C.
2. Java Programming Language Review. A study of linear data structures in Java, including stacks, queues, lists, sets, bags, dictionaries, and priority queues.
3. Trees in Java. Coverage of binary search trees, heaps, and balanced trees, along with different data structure implementations based on trees.
4. Hash Tables in Java. An analysis of hash functions, collision resolution techniques, and the use of hash tables to implement various data structures.
5. Graphs in Java. Exploration of both directed and undirected graphs, including depth-first and breadth-first traversals and topological sorting.

Assessment
Continuous assessment during some laboratory sessions and on computer examinations.

Course Syllabi

Lecturer
Dr. José E. Gallardo
jegallardo@uma.es
Room 3.2.50

Course Syllabi

CODE 205

NAME Statistics Methods

Credits 6 ECTS

Period Fall Semester

Course Specifications

Classes will be held in the classroom, including problem-solving sessions both with and without the use of computers. Individual effort is expected from students.

Objectives and contents

The course has a dual focus: on the one hand, to describe the usual concepts and procedures in statistical analysis, probability calculation, and hypothesis testing, by proposing and solving small-scale problems; and on the other hand, to code these procedures using the R language, which allows for solving problems involving large volumes of data. It is recommended to learn the language in the early stages of the educational process to enable problem-solving at both levels.

Contents

1. Descriptive Statistics: Descriptive Statistics for One and Multiple Variables.
2. Statistical Models: Simple and Multiple Linear Regression. Statistical Modelling.
3. Time Series: Time Series Decomposition.
4. Probability Calculation: Probability. Random Variables and Distributions. Notable Distributions.
5. Statistical Inference: Point Estimation and Confidence Intervals. Parametric Hypothesis Testing. Non-Parametric Estimation.

Assessment

The evaluation for the first ordinary call consists of:

- *Continuous Assessment*: This involves completing activities proposed throughout the course. These activities will be graded from 0 to 10 and will account for 40% of the final grade.
- *Final Exam* : This will involve solving questions and problems. The exam will be graded from 0 to 10 and will account for 60% of the final grade.

For the remaining calls, the evaluation will consist of a single exam.

Course Syllabi

Lecturer
Dr. Iván Atencia Mc.killop
iatencia@ctima.uma.es
Room 2.2.16

Course Syllabi

CODE 305

NAME Intelligent Systems

Credits 6 ECTS

Period 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 is a final written exam and mandatory tasks to be handed out through the virtual campus. Also, there are mandatory programming exercises, to be coded in Python. Therefore, programming skill is a prerequisite. Active class participation and other optional activities give extra marks.

Lecturer
Dra. Aurora Ramírez Quesada
aurora.ramirez@uma.es
Room: 3.2.8

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

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

Dr. Miguel Ángel Molina
miguelangel@lcc.uma.es
Room 3.3.2.I.

Course Syllabi

CODE 106

NAME Calculus for Informatics

Credits 6 ECTS

Period Spring 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 2.2.19

Course Syllabi

CODE 107

NAME Algebraic Structures

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

First Call

The course is divided into two thematic blocks, with equal weight for obtaining the successive grades and their transfer to the official records:

- Block 1. Consists of the topics:
  - Algebraic structures: groups, rings and fields. Coding theory.
  - Cardinality.
  - Partially ordered Sets, lattices and Boolean algebras.
- Block 2. The contents will be those related to linear algebra:

- Linear equations systems.
- Vector Space.
- Linear transformations.
- Diagonalization.
- Euclidean Vector Space.

In the first ordinary call, the subject will follow a continuous evaluation. For each block, the following will be done:

- A midterm exam of the contents of the block, to be carried out in the lecture room, with a weighting of 80%.
- Questionnaires or knowledge tests of a theoretical/practical nature, to estimate the evolution of the students, in the middle of the block. This questionnaire will have a weighting of 10%.
- An individual practice to be done with a computer, during lectures. This practice has a weighting of 10%.

This weighting results in an independent rating for each block.

If a grade greater than or equal to 3.5 is achieved in both blocks, the average of both grades will be calculated. If said average is equal to or greater than 5, this will be the grade that is transferred to the official records in the first call.

On the date indicated by the School for the exam of the first call, two partial exams will be held, one for each block. Each student has to take the midterm corresponding to each block where he or she has not obtained at least a 5.

The grade obtained in these midterms will replace the one achieved through continuous evaluation (midterms, questionnaires and practices) in the corresponding block, since these last midterms represent the final milestone of the continuous evaluation of the course.

After completing these partial exams, the evaluation will follow the same scheme as in the continuous evaluation.

Those students who, having not passed the course through continuous assessment, do not take the corresponding partial exams of the first call, on the date set by the School, will appear in the official records as "not presented."

Second Call

If the first call is not passed, on the date proposed by the school a single exam will be taken on the complete content of the course, the grade of which will be the one that is transferred to the official records.

Lecturer

Dra. Mª Carmen Fernández Gago
mcgago@uma.es
Room 2.2.21

Course Syllabi

CODE 108

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 focused on the aspects of code management, version control, quality assurance, testing and modelling. We cover the necessary theoretical aspects, but the course is very practical. Therefore, we will introduce the concepts, but then always apply them using specific techniques, tools, and standards. Our goal is to provide students with a good understanding of the pillars of software development that will provide them with the necessary skills to successfully take courses on other specific areas of software engineering.

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

Course Syllabi

CODE 109

NAME Advanced Programming I

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 the Java programming language.

Contents

1. Introduction in Java to Object Oriented Programming Fundamentals: classes, objects, methods, messages, composition, variables, predefined types.
2. Inheritance and interfaces. Single and multiple inheritance, interface implementation, abstract classes, static and dynamic binding.
3. Exception management. Error handling with exceptions. Throwing exception, definition of tailored exception classes.
4. Basic predefined Java classes. System basic classes (java.lang), system utility classes (java.util), input/output in java (java.io).
5. Collections. Generic programming in Java. Predefined Java collections (lists, sets) and maps.

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

Course Syllabi

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 Virtual Campus 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/
Room 2.116D (Escuela de Ingenierías)

Course Syllabi

CODE 206

NAME Foundations of Artificial Intelligence

Credits 6 ECTS

Period Spring 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 three written exams: two midterm exams and a final exam. The grades of the two midterm exams are 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 Python. Therefore, programming skill is a prerequisite. Active class participation and other optional activities give extra marks.

Lecturer
Dr. José Luis Pérez de la Cruz
ccia@uma.es
Room: 3.2.24

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

Course Syllabi

CODE 207

NAME Advanced Programming II

Credits 6 ECTS

Period Spring Semester

Course Specifications

Lectures where theoretical concepts, problems and algorithms are explained in the classroom whereas, in the laboratory, every student puts in practice these concepts resolving well-defined problems following the guidelines of the teacher. The course follows a practical approach using the Scala and Java programming languages.

Objectives and contents

The main objectives are to learn functional and parallel programming in contrast to traditional programming models. Classical problems are shown as well as their solutions following different approaches and the syntactic and semantics resources provided by Scala. A last objective is to learn how to create programs with GUI (Graphical User Interfaces) using a well-founded methodology.

Contents

1. Functional programming.
2. Concurrency: main concepts.
3. Concurrency: communication and synchronization.
4. Event driven programming.

Assessment

There are 3 midterm exams on i) functional programming, ii) semaphores and iii) monitors/locks, each with a value of 30%. Each exam must be passed with a minimum of 4 over 10. A passed exam is kept (no need to repeat it) in any ordinary call of the course. The remaining 10% of the mark corresponds to a home task on Java GUI.

Lecturer

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

Course Syllabi

CODE 208

NAME Networks and Services

Credits 6 ECTS

Period Spring Semester

Course Specifications

The course includes lectures, problem-solving sessions, individual work, and laboratory sessions.

Objectives and contents

This course provides an overview of computer networks, starting with the basics of networking, including their structure and how networks are organized.

Contents

It introduces layered models, such as the Internet models, and the standards that define network protocols and operations. The discussion of network access and link control includes the services and functionalities of the link layer, focusing on multiple access networks, local area network (LAN) technologies, and other specialized networking technologies. Network Interconnection Protocols and Internet Services cover VLANs, and IP protocols (IPv4 and IPv6), including Internet routing. It also explores transport layer services, discussing protocols like UDP and TCP, along with network programming with sockets APIs. Additionally, it presents classical Internet services such as DNS, the World Wide Web, and email among others, and advanced Internet services, such as cloud computing, multimedia, and other cutting-edge elements.

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 score.

Lecturer

Dra. Lidia Fuentes
lfuentes@uma.es
Room 3.2.8

Course Syllabi

CODE 209

NAME Operating Systems

Credits 6 ECTS

Period Spring Semester

Course Specifications

The course includes lectures about fundamental concepts involved in the internal design of any Operating System. It also includes laboratory work about programming complex software services on the POSIX of the UNIX family of operating systems.

It requires deep and strong knowledge of the hardware 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).

Note for Erasmus students

If you come from a **non Computer-Science or Information Technology degree**, you are advised to **not choosing this subject**.

If your Erasmus supervisor recommends you to take it, please refer him or her to contact the lecturers (guille@ac.uma.es) to discuss the suitability of this subject. We have a recurring series of badly advised students that fail to understand the theory of the subject because of the lack of previous knowledge.

Just being an experienced programmer is not enough as a base to approach this subject.

Objectives

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:

- Many abstractions offered by programming languages (like threads, processes, files, network sockets, memory allocation, etc.) are not part of the programming language. Instead, the OS implements them and the languages just call the through system calls. This is why all programming languages offer the same abstractions.
- Security is also implemented in the Operating System. The concept of user and access permissions does not exists in the hardware but created and managed by the code of the OS. That includes controlling the access to the hardware resources by the programmer.

With these concepts in mind, you will understand that High Level Languages are only one more layer on top of OS software. You will become a better programmer understanding that any language is just generating code that uses the common OS implementation of resources. In this sense, you will be able to focus on management of resources more than on the language itself.

Contents

1. Processes and threads.
2. Process scheduling.
3. Memory management.
4. Input/output subsystem.

Laboratories

5. The UNIX programming environment: Programming on the POSIX standard.
6. Low-level binary files.
7. Communication through the Operating System (shared memory and queues).
8. Signals and processes.
9. Graceful termination of service programs.

Assessment

For the standard student, continuous evaluation of the progress is compulsory. There will be NO FINAL exam at the end of the course.

If the student misses an intermediate exam laboratory work (with exception of properly documented causes like medical reasons, severe illness or decease of close relatives, etc.) the final score will be NOT EVALUATED.

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.
- 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.

Lecturers

Dr. Ricardo Quislant: (theory sessions)
quislant@uma.es
Room 2.2.25

Dr. Guillermo Pérez (laboratory sessions)
gperez@uma.es
Room 2.2.34

Course Syllabi

CODE 210

NAME Automata Theory and Formal Languages

Credits 6 ECTS

Period Spring 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