ETSI Informática

University of Málaga

Foundations of Computer Science

Course Guide
2023-2024

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 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 **2023-2024**. 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:

M.ª Magdalena de la Chica y Lorena Caro
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

E.T.S.I. INFORMÁTICA
CURSO 2023 - 2024

septiembre 2023

L	M	X	J	V	S	D
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	1
2	3	4	5	6	7	8

octubre 2023

L	M	X	J	V	S	D
25	26	27	28	29	30	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

noviembre 2023

L	M	X	J	V	S	D
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	1	2	3
4	5	6	7	8	9	10

diciembre 2023

L	M	X	J	V	S	D
27	28	29	30	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31
1	2	3	4	5	6	7

enero 2024

L	M	X	J	V	S	D
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4
5	6	7	8	9	10	11

febrero 2024

L	M	X	J	V	S	D
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	1	2	3
4	5	6	7	8	9	10

marzo 2024

L	M	X	J	V	S	D
26	27	28	29	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31
1	2	3	4	5	6	7

abril 2024

L	M	X	J	V	S	D
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5
6	7	8	9	10	11	12

mayo 2024

L	M	X	J	V	S	D
29	30	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2
3	4	5	6	7	8	9

junio 2024

L	M	X	J	V	S	D
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
1	2	3	4	5	6	7

julio 2024

L	M	X	J	V	S	D
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4
5	6	7	8	9	10	11

septiembre 2024

L	M	X	J	V	S	D
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

Lectivo Instructional Day	No lectivo No lectures	Festivos Public holiday	Inicio/fin de semestre Start/End of semester	Exámenes ordinarios Ordinary Exams
Exámenes Extraordinarios (lectivos) Extraordinary Exams (lecture days)	Actividades culturales - Hackers Week/Days (lectivos) Cultural Activities - Hackers Week/Days (lecture days)			

Periodos de docencia y exámenes / Teaching and Exams Periods

	Inicio/Start	Fin/End	
Primer cuatrimestre	12/09/2023	22/12/2023	Lectures 1st Semester
Segundo cuatrimestre	16/02/2024	05/06/2024	Lectures 2nd Semester
Exámenes conv. Extraordinaria	06/11/2023	24/11/2023	Extraordinary exams (instructional days
Exam. 1ª Ord. - 1º Cuatri.	08/01/2024	26/01/2024	1st Semester 1st Exams
Exam. 2ª Ord. - 1º Cuatri.	06/02/2024	15/02/2024	1st Semester Exams 2nd Exams
Exam. 1ª Ord. - 2º Cuatri.	06/06/2024	26/06/2024	2nd Semester 1st Exams
Exam. 2ª Ord. - 2º Cuatri.	05/07/2024	16/07/2024	2nd Semester 2nd Exams

Días Festivos / Public Holydays

	Inicio/Start	Fin/End
Nuestra Señora de la Victoria	08/09/2023	
Día del Pilar	12/10/2023	
Todos los Santos	01/11/2023	
Día de la Constitución	06/12/2023	
Inmaculada Concepción	08/12/2023	
Navidad/Christmas Holidays	23/12/2023	07/01/2024
Santo Tomás de Aquino	29/01/2024	
Patrón de Informática	30/01/2024	
Día de Andalucía	28/02/2024	
Hackers Week (lectivo)	04/03/2024	08/03/2024
Semana Santa (Easter Holidays)	22/03/2024	01/04/2024
Fiesta del Trabajo	01/05/2024	

Program Shedule

Fall Semester (70.5 ECTS)

Code	Name (ECTS)
101	Electronic Foundations (6)
102	Physics Foundations of Computer Science (6)
103	Introduction to Programming (6)
104	Discrete Mathematics (6)
108	Introduction to Software Engineering (6)
201	Analysis and Design of Algorithms (6)
202	Databases (6)
203	Computer Organization (6)
204	Data Structures (6)
205	Automata Theory and Formal Languages (6)
305	Intelligent Systems (6)
920	Computational Intelligence (4.5)

Spring Semester (54 ECTS)

Code	Name (ECTS)
106	Calculus for Informatics (6)
107	Algebraic Structures (6)
109	Advanced Programming I (6)
110	Computer Technology (6)
206	Introduction to Software Engineering (6)
207	Systems Programming and Concurrency (6)
208	Networks and Distributed Systems (6)
209	Intelligent Systems (6)
210	Operating Systems (6)

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 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 3.2.19

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

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 assessment of the course for the first call will follow a continuous evaluation process consisting of handed-in activities and exercises, with or without computer support, questionnaires, or partial exams. The number of these activities will depend on the organisation and evolution of the course. They could also depend on the criteria set up by the management of the School and the real size of the group of the course.

The weight of the assessment during the teaching period of the course will be 73%. The remaining 27% will be assessed in a partial exam that will take place on the date scheduled by the School for the first call exam. Only the mark of the students who take this partial exam will be transferred to their official records for marks.

Second Call
An exam will be held on the date scheduled by the School with the parts of the course not passed in the first exam.

Other 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 2.2.21

Course Syllabi

CODE 108

NAME Introduction to Software Engineering

Credits 6 ECTS

Period Fall 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 cover the concepts, processes, techniques, tools, and standards that affect the development of a software project with a focus on the aspects of code management, version control, testing and quality assurance. 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 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
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

1. Functional Programming.
2. Linear data structures: stacks, queues, and lists.
3. Trees: search trees, heaps, and balanced trees.
4. Hash tables: hash function and collision resolution techniques.
5. 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

Course Syllabi

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 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 are three written exams: two midterm exams and a final exam. The grade 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.

Lecturers

Dr. Enrique Domínguez Merino
enriqued@uma.es
Room: 3.2.7

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

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. Preliminary Concepts. Cardinality
2. Lattices and Boolean Algebra
3. Groups, Rings and Fields
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

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.

Students who do not pass the course following this procedure could do the final exam on the date scheduled by the School.

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 2.2.21

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 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. Javier Cámara Moreno
jcamara@uma.es
Room 3.3.2

Course Syllabi

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

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

Course Syllabi

CODE 209

NAME Intelligent Systems

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. Enrique Domínguez Merino
enriqued@uma.es
Room: 3.2.7

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 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