
DEGREE CURRICULUM

# INDUSTRIAL COMPUTING

Coordination: GUIRADO FERNÁNDEZ, FERNANDO

Academic year 2019-20

# INDUSTRIAL COMPUTING 2019-20

## Subject's general information

<table>
<tr><td>Subject name</td><td>INDUSTRIAL COMPUTING</td></tr>
<tr><td>Code</td><td>102129</td></tr>
<tr><td>Semester</td><td>2nd Q(SEMESTER) CONTINUED EVALUATION</td></tr>
<tr><td>Typology</td><td>
<table>
<tr><th>Degree</th><th>Course</th><th>Character</th><th>Modality</th></tr>
<tr><td>Bachelor's Degree in Automation and Industrial Electronic Engineering</td><td>3</td><td>COMPULSORY</td><td>Attendance-based</td></tr>
</table>
</td></tr>
<tr><td>Course number of credits (ECTS)</td><td>6</td></tr>
<tr><td>Type of activity, credits, and groups</td><td>
<table>
<tr><th>Activity type</th><th>PRALAB</th><th>PRAULA</th><th>TEORIA</th></tr>
<tr><th>Number of credits</th><td>0.4</td><td>2.6</td><td>3</td></tr>
<tr><th>Number of groups</th><td>4</td><td>1</td><td>1</td></tr>
</table>
</td></tr>
<tr><td>Coordination</td><td>GUIRADO FERNÁNDEZ, FERNANDO</td></tr>
<tr><td>Department</td><td>COMPUTER SCIENCE AND INDUSTRIAL ENGINEERING</td></tr>
<tr><td>Teaching load distribution between lectures and independent student work</td><td>30% lecture<br>70% autonomous work</td></tr>
<tr><td>Important information on data processing</td><td>Consult this link for more information.</td></tr>
<tr><td>Language</td><td>Idioma Percentatge d'ús<br>Castellà 0.0<br>Català 90.0<br>Anglès 10.0</td></tr>
<tr><td>Distribution of credits</td><td>20% theoretical content<br>30% classroom activities<br>50% laboratory</td></tr>
<tr><td>Office and hour of attention</td><td>Email to the professor</td></tr>
</table>

# INDUSTRIAL COMPUTING 2019-20

| Teaching staff | E-mail addresses | Credits taught by teacher | Office and hour of attention |
|---|---|---|---|
| GUIRADO FERNÁNDEZ, FERNANDO | fernando.guirado@udl.cat | 0 | |
| PARDO CASANOVAS, VALENTI | valenti.pardo@udl.cat | 7,2 | |

## Subject's extra information

Course taught in the second semester of the third year.

Is within the specific training modules.

It is recommended to have a user-level knowledge of operating systems and programming.

## Learning objectives

- User-level knowledge of Linux operating system
- Know how to use the commands associated with the file system in LINUX
- Understand the concept of process and how to use the associated tools in Linux
- Representation of computational problems by pseucode
- To know the programming language C
- Developing small applications in C language

## Competences

**Strategic competencies of the University of Lleida**

UDL3 - Domain of Information Technology and Communication.

**Specific skills of the program**

GEEIA3. Basic knowledge of the use and programming of computers, operating systems, databases and computer programs with applications in engineering.

GEEIA19 - Applied knowledge of industrial computing and communications.

**Transversal competences of the degree**

EPS1 - Ability to solve problems and develop and defend arguments in their area of study.

## Subject contents

1. Operative System LINUX
    1. The file system structure
    2. The file system Permission
    3. Process manegement
2. Language C Programming

1. The GNU compiler
2. Compiler directives
3. Data types
4. Variable declaration
5. Operators
6. Basic programming structures
7. Functions
8. Data estructures
9. Pointers and dynamic memory management.

## Methodology

The course is composed by:

- Lecture classes: The main theory for the course will be presented.
- Individual exercices: Some exercices, proposed by the professor, have to be developed and presented individually.
- Laboratory practices: Some practical work will be done using the computers present in the laboratory.

## Development plan

| | **Development** |
|---|---|
| Week 1<br>(4h Lecture class / 6h Autonomous work) | Presentation and contents summary |
| | Introduction to the Operative Systems. LINUX operative System - Introduction to the file system |
| Week 2<br>(4h Lecture class / 6h Autonomous work) | LINUX operative System - Introduction/File system |
| | LINUX operative System - File system |
| Week 3<br>(4h Lecture class / 6h Autonomous work) | LINUX operative System - Permissions |
| Week 4<br>(4h Lecture class / 6h Autonomous work) | LINUX operative System - Process status |
| Week 5<br>(4h Lecture class / 6h Autonomous work) | Computational solving problems |
| | C Programming Introduction |
| Week 6<br>(4h Lecture class / 6h Autonomous work) | C Programming Introduction |
| Week 7<br>(4h Lecture class / 6h Autonomous work) | Error control and deputarion |
| | Structured programming - Functions - Libraries |
| Week 8<br>(4h Lecture class / 6h Autonomous work) | Structured programming - Functions - Libraries |
| | *Partial exam doubts session* |
| Week 9 | **Partial Exams** |
| Week 10<br>(4h Lecture class / 6h Autonomous work) | Arrays - Vectors |

| | |
|---|---|
| Week 11<br>(4h Lecture class / 6h Autonomous work) | Data structures |
| Week 12<br>(4h Lecture class / 6h Autonomous work) | Pointers |
| Week 13<br>(4h Lecture class / 6h Autonomous work) | Dynamic memory management |
| Week 14<br>(4h Lecture class / 6h Autonomous work) | Dynamic memory management |
| Week 15 | *Doubts session* |

## Evaluation

The grade has two parts: Exams and Practice

The mark of the exam is 30% of the total mark of the subject. There will be two exams one for each part, which will have the same weight and have no minimum mark.

The note of practice is 70% of the total mark of the subject.

The practices of the course are:

- PRA1. · Linux Filesystem
- PRA2. Linux · Permits
- PRA3. · Linux Processes
- PRA4. C · flow control
- PRA5. C · Structured programming: functions and libraries
- PRA6. C · Vectors and random numbers
- PRA7. · C Data Structures
- PRA8. · · C pointers Dynamic memory

Requirements / Features practices:

Practices are compulsory and non-recoverable.
In order to be evaluated, the practices must pass a validation test that will not take any mark.
They have no minimum mark.
Practices can be made in pairs.

## Bibliography

**Bibliografia recomanada**

Blanco, Jaime. Linux/Ubuntu : curso de iniciación. Inforbook's, DL 2006

Dalheimer, Matthias Kalle. Guía de referencia y aprendizaje Linux. Anaya Multimedia, cop. 2006, Edición 2ª ed.

H.M. Deitel and P.J. Deitel. ComoProgramar en C/C++. Prentice-Hall, segunda edición, 2002.

B.W. Kernighan and D.M. Ritchie. Ellenguaje de programación C. Prentice-Hall, segundaedición, 1991.

F.Xhafa; P. Vázquez, J. Marco, X. Molinero and A. Martín. Programación en C++ paraingenieros. Paraninfo, 2006.

## Adaptations to the contents due to COVID-19

Update test Adaptations to the contents -ds. 25 d'abr. 2020 10:22:36-

## Adaptations to the methodology due to COVID-19

Update test Adaptationsto the methodology -ds. 25 d'abr. 2020 10:22:36-

## Adaptations to the development plan due to COVID-19

Update test Adaptations to the development plan -ds. 25 d'abr. 2020 10:22:36-

## Adaptations to the evaluation due to COVID-19

Update test Adaptations to the evaluation  -ds. 25 d'abr. 2020 10:22:36-