Universitat de Lleida

DEGREE CURRICULUM

PROGRAMMING AND COMMUNNICATIONS I

Academic year 2015-16

PROGRAMMING AND COMMUNICATIONS I 2015-16

Subject's general information

Subject name	Programming and Communnications I
Code	102133
Semester	1r Q Continued Assesment
Typology	Optative
ECTS credits	6
Groups	1
Theoretical credits	3
Practical credits	3
Office and hour of attention	To arrange with the teacher
Department	Computer Science and Industrial Engineering
Modality	Presencial
Important information on data processing	Consult this link for more information.
Language	English
Degree	Degree in Automation and Industrial Electronic Engineering
Office and hour of attention	To arrange with the teacher
E-mail addresses	ramon@diei.udl.cat

Ramón Béjar Torres

Subject's extra information

Previous Requirements

The student must have taken and passed the subjects from the "basic training" module and the industrial computing subject of the "common training" module.

For students who come from other university degrees, you must have completed subjects covering basic knowledge about fundamentals of computer programming, have basic knowledge about the Linux operating system and electronic circuits and digital sensors.

Learning objectives

Learning Goals

Expected learning outcomes related to the strategic transversal competences:

- The student is able to learn and work with technical documentation in English on programming languages ??on Linux environment (Competences UdL2, EPS4).

- The student is able to work in teams to carry out the development of a computer system composed of different subsystems in which there to apply knowledge from different fields (Competence EPS9).

- The student is able to prepare presentations in English to show aspects main programs developed for other equipment engineers can understand their solutions. (Competences UdL2, UdL3).

- The student is able to generalize basic algorithmic schemes to apply them in different contexts and problems from the ones initially seen (Competence EPS4).

Expected learning outcomes linked to specific competencies:

- The student is able to understand the basic features of the functioning of operating systems multitasking, multiuser based on the Linux kernel (Competences GEEIA3 and GEEIA34).

- The student is able to integrate the knowledge of circuits, sensors and processes industrial on knowledge of computer programming for address the full development of small automated systems monitoring / control software based processes on Raspberry Pi computers or microcomputers (Competences GEEIA3, GEEIA28 and GEEIA34).

Competences

Strategic competences UdL:

- UdL2. Knowledge of an foreign language.
- UdL3. Knowledge of ICT.

Transversal competences EPS:

- EPS4. Have the learning abilities needed to start superior studies or improve the academic learning with a certain autonomous degree.
- EPS9. Ability to work in teams, both as a unidisciplinary and multidisciplinary.

Specific competences GEEIA:

- GEEIA3 Basic knowledge on using and programming computers, operating systems, databases and software with applications in engineering.
- GEEIA28. Applied knowledge of industrial computing and communications.
- GEEIA-EPS34. Knowledge of the fundamentals of computer systems and applications.

Subject contents

- Introduction
- Variables, expressions and statements
- Conditional execution
- Functions
- Loops and Iterations
- Strings
- Files
- Lists, Dictionaries and Tuples
- Filtering Information
- RaspberryPI setup and configuration
- Getting data from sensors with the RaspberryPI GPIO

Methodology

Learning activities
Face to face activities (40%): The percentages associated to each one of the activities are computed over 100%

- Master class (42,5%)
- Problems (25%)
- Laboratory (25%)
- Tests and evaluation (7,5%)

Autonomous work (60%): The percentages associated to each one of the activities are computed over 100%

- Working on the mandatory programming assignments (80%)
- Solving small programming problems (20%)

Development plan

Each week the student will have two hours of mainly master class, although some programs will be seen that students will have to begin working with to adapt them to similar problems.

The other two hours of class will be for working in the programming laboratory, working with both PCs and the micro-computer Raspberry Pi, to work on solving the problems or programming projects that have to be solved in teams 2/3 students.

Evaluation

3 separate items:

- Mandatory programming assignments: 2 programming projects (80% weight).
- Small exercises (during the course) to assess some important milestones, max 4, (20% weight).
- Oral presentations for individual validation of the programming assignments.

Bibliography

On-line resources.

Python: http://docs.python.org/2.7/

Raspberry Pi: http://www.raspberrypi.org/

Bibliography:

- Mark Lutz. Learning Python 4th Edition. OReilly - 2009.
- Zed A. Shaw. Learn Python the Hard Way - 2014 http://www.souravsengupta.com/int2pro2014/python/LPTHW.pdf
- Learn Raspberry Pi Programming with Python - Wolfram Donat - Apress. 2014. http://www.allitebooks.com/learn-raspberry-pi-programming-with-python/
- Raspberry Pi: A Quick-Start Guide, 2nd Edition -Maik Schmidt - The Pragmatic Programmers, 2014
- Raspberry Pi Cookbook - Simon Monk - O'Reilly- 2014
- Raspberry Pi Home Automation with Arduino - Andrew K. Dennis - Packt Publishing, 2013

Some free on-line books for learning python:

- Dive into python. http://www.diveintopython.net/
- A Byte of Python - Una mica de Python. http://moiatgit.github.io/byte_of_python_120.cat/

The free on-line resources are enough to follow this subject, but if you want to have a good book for developing and understanding many classes of problems and programs on the RsPI, choose the *Raspberry Pi Cookbook*