

# Laboratórios de Computadores: Apresentação

## Computer Labs: Introduction

### 2º MIEIC

Pedro F. Souto (pfs@fe.up.pt)

September 16, 2019

# Staff

- ▶ Sara Fernandes (???@fe.up.pt)
- ▶ Nuno Paulino (nmcp@fe.up.pt)
- ▶ Pedro Silva (pmms@fe.up.pt)
- ▶ Pedro Ferreira do Souto (pfs@fe.up.pt)
- ▶ Rui Grandão (ruirocha@fe.up.pt)

# Context

- ▶ I/O devices are an integral part of a computer
  - ▶ Without them, a computer would be like a heater
- ▶ The relevance of I/O devices has increased significantly with the deployment of an increasing number of embedded systems and the emergence of the Internet of Things
  - ▶ An "embedded system" is a computer-based system that is part of an "object" and interacts with the physical-world.  
E.g. a car's ABS, a washing machine controller, ...
- ▶ However, programming of I/O devices requires specific knowledge and techniques

# Objectives

*This course aims to endow students with the knowledge and the skills required to:*

- 1. Use the programmatic interface of the most common computer I/O devices;*
- 2. Develop low-level/system-level and embedded programs*
- 3. Use software tools typical of large programming projects*

# Prerequisites

- ▶ **Programação**

- ▶ You'll program a lot, mostly in C
  - ▶ but also in assembly
  - ▶ ... for the IA-32 architecture

- ▶ **Microprocessadores e Computadores Pessoais**

- ▶ Arquitectura de Computadores

- ▶ If you have **not** completed Programação, please consider to drop LCOM, if you do not have enough time to dedicate to programming

# Syllabus

I/O devices and their programming techniques

C programming

Programming tools

# Method

## Learn by doing

“I hear, I forget. I see, I remember. I do, I understand”

## Several short lab assignments

- ▶ Each focusing on one I/O device
- ▶ Some of them take only one lab class, others take two lab classes
- ▶ Requiring a preparation of about 6 hours per lab class (not including lectures or lab classes)

## One integration project

- ▶ Must use
  - ▶ at least 3 different I/O devices
  - ▶ interrupts
- ▶ Expected effort: about 9 hours per week (during 5 weeks)

**Note:** Both lab assignments and project should be done in groups of 2 students.

# Project Examples

- ▶ Games (video, timer, keyboard and mouse)
- ▶ Two user games (video, timer, keyboard and serial port)
- ▶ Electronic calendar (video, keyboard, mouse, RTC and timer)
- ▶ Music composer/player (video, keyboard, mouse and timer)
- ▶ Text editor (video, keyboard, mouse, timer and RTC)
- ▶ Typing tutor (video, keyboard, mouse, timer)
- ▶ File transfer between PCs (video, keyboard, serial port)
- ▶ Chat between PCs (video, keyboard, serial port)
- ▶ Video player (video, keyboard, mouse, timer and RTC)
- ▶ Drawing/painting program (video, keyboard, mouse, timer, RTC and serial port)



# Since last year: The LCOM Framework (LCF)

- ▶ Thanks to Pedro Silva, who designed and implemented it
- ▶ It allows us to provide you
  - Helper functions, that will allow you to more easily debug some common and time consuming errors
  - Test cases so that you can be more confident that your code is working as requested
- ▶ Furthermore, the LCF will allow us to automatically grade your code
  - ▶ Actually, we will still need to do some grading "manually", but we expect it to be less than 10%

# New this year: a lab assignment on real HW

New lab assignment on an embedded systems/IoT platform

- ▶ Other lab assignments are on the PC
- ▶ Other lab assignments use a virtual machine

NOT DECIDED YET depends on availability of resources

# Work Load

- ▶ LCOM has 6 ECTS, i.e. about 160 hours
  - ▶ Assuming 1 ECTS equal to 27 hours
    - ▶ Check out the [European Credit Transfer and Accumulation System \(ECTS\)](#)
  - ▶ If you share the load with your team-mate, this should not be a problem.

Unit	Hours/Week	No. Weeks	Total
Lectures	2	13	26
Labs classes	3	12	36
Prep. L0	5	1	5
Prep L2-L5	6	8	48
Proj.	9	5	45
Total			160

- ▶ If there is a lab on real HW, it will take 2 lab classes
  - ▶ Its preparation it will be approximately 6 hours per week
  - ▶ The project load will be reduced by approximately 12 hours.

# Bibliography and Other Resources

- ▶ PC HW is well documented on several books and online resources
- ▶ Book mentioned in SIFEUP  
***Mazidi, Muhammad**, The 80x86 IBM PC and Compatible Computers: Assembly Language, Design and Interfacing, 4th Ed., Prentice-Hal*

Note that it does not cover all the subjects, and that, on the other hand, it has a lot more material than needed for this class.

# Grading (1/2)

**Labs (Labs)** Of the 4 graded labs, we will use 3 for computing this component

- ▶ For a final grade of 19 or 20, must use the grade of the mouse lab

## Project (FP)

**Course Participation (CP)** covers lab preparation/lecture participation proper (see slide at the end)

**Programming Test (PT)** individual test similar to a lab, but much smaller, on a pseudo-device

- ▶ **Minimum score of 8 required for approval**
- ▶ Can be repeated during exam period
  - ▶ Maximum score in the **repeated PT** is 8
  - ▶ If you got the minimum score, you cannot repeat it

**Formula**  $0.4 \text{ Labs} + 0.1 \text{ CP} + 0.1 \text{ PT} + 0.4 \text{ FP}$

**Embedded Systems Lab** if we implement it this year, it will be 10% of the final grade

- ▶ In this case, we will make some adjustments to the formula above (within the next 2-3 weeks).

## Grading (2/2)

- ▶ All grades are individual
  - ▶ Even though labs and projects should be done in groups
- ▶ Lab/project grades are normalized for groups of 2
- ▶ The grade of each group member is obtained by applying a factor to the group's grade
  - ▶ The factor depends on the member's **contribution** mostly **participation** also to the final result
    - ▶ In principle, if you choose to be the only group member, the factor's value will be 1.
  - ▶ We will use a piecewise linear function to determine the value of the factor
    - ▶ Likely "breakpoints" are: (0,0), (0.33,0.8), (0.5,1), (0.8, 1.1) and (1, 1.1)
- ▶ Each member of the group must fill a Google form with its own self-assessment (contribution and participation) of each of the labs/projects submitted by the end of the day of the respective deadline

# Labs Grading

- ▶ You must submit your code via SVN
  - ▶ This year, the deadline is about 10 minutes after the end of your last lab class for the lab.
- ▶ You must also fill a self-evaluation form for each lab.
- ▶ Grading will be mostly automatic
  - ▶ It will allow us to publish the lab grades much faster
- ▶ We will provide you a subset of the test cases used for grading, to prevent you from getting low marks, just because you did not understand the requirements
- ▶ Do not ask us for a submission extension
  - ▶ We use only 3 of the 4 graded labs for computing the labs grade component

# Programming Test

- ▶ It is an individual test similar to a lab, but smaller, and may be on a device that is not real
- ▶ You will have access to, at least, the handouts of both the lectures and the labs
- ▶ Grading will be fully automatic
  - ▶ We will not consider criteria that we are unable to automate
- ▶ You must have a minimum score of 8 valores to pass in LCOM
- ▶ You can repeat the test during the exam period, but only if you failed because of the PT
  - ▶ Maximum score in the **repeated PT** is 8
  - ▶ This ensures that you cannot have a final grade higher than that of any student that got the same (aggregate) grade on all other components, and was not allowed to repeat the PT, because s/he had passed



# Final Project Grading (1/2)

Execution: 35% + 10%

- ▶ 10% for demo in the last lab class

Code: 20%

- ▶ Structure and Modularity
- ▶ Documentation (use Doxygen)
- ▶ Readability
  - ▶ Names and comments
  - ▶ Indentation
- ▶ Compilation warnings

Final Report: 20%

- ▶ Summary of what is and what is not implemented;
- ▶ Usage instructions (with images)
- ▶ Description of the program's architecture
- ▶ **Relevant** implementation aspects (grades above 18)
- ▶ Function call diagram

Video: 5% A short video with a demo of your project

Tools: 5% (SVN) (We expect you to commit to the SVN repository at least once a week, and to log messages then)

Project Specification: 5%

# Final Project Grading (2/2)

- ▶ To the grade obtained by applying the above criteria, we'll apply:

## Difficulty Factor

- ▶ number and type of I/O devices
- ▶ features used of the I/O devices
- ▶ I/O techniques used (interrupt vs. polling)
- ▶ use and extent of assembly programming

## Originality Factor

## Team Management Factor

- ▶ load share among group members

## Marketing Bonus

- ▶ of 1 valor for the participation in the Semana Profissão Engenheiro (SPE), sometime in March 2018
- ▶ in recent years, we have selected 3 or 4 projects per year

# Final Project Milestones

**Project proposal:** Beginning of the 7th lab class (week starting 11th November)

- ▶ Half to one page description of the functionalities desired, of the devices used and their role in the program
- ▶ Must be rewritten in class, if the instructor does not accept it

**Project specification:** Beginning of 8th lab class (week starting 18th November)

- ▶ Refinement of the proposal, identifying the modules to implement, their functionality and API.
- ▶ Should include planning of the project
- ▶ Must be rewritten in class, if the instructor does not accept it

**First demo:** At the last lab class of the semester.

**Project submission:** January 6th, 20:00

**Project presentation:** January: 8th to 10th

# TEs Grading

Similar to the grading of other students

- ▶ I.e. it is based on lab assignments, programming test and project

Except that student may demo:

Labs

Project

at a time different from their lab class

In that case The student must send me (pfs@fe.up.pt) and email **at least 7 days in advance** the previous week to arrange for a date and time

# “Época Especial”

► There is none.

“Melhoria de nota”

► Next year.

# Special Evaluation

**IMPORTANT** Students wishing to:

1. be assessed as TE's
2. use their labs/project/PT (positive) grades from 2017/2018 or 2018/2019

must fill [this Google form](#) by the end of this week, i.e. 2019-09-22.

**IMPORTANT** Please note that by choosing this option, you may have to work alone in some or all grading components

- ▶ If you do not want to work alone, it will be up to you to create a group (together with other students) or find a group that accepts you as a new member.

# Academic Integrity

- ▶ The UP and we take this issue very seriously
  - ▶ Check the [Despacho do Reitor Nº 08/09/2011](#)
  - ▶ We believe that the majority of you follow the rules
- ▶ You are allowed to discuss the labs
  - ▶ For each lab and for the project, there will be a discussion forum on Moodle
- ▶ However, all code submitted should be either:
  - ▶ Developed by the group members
  - ▶ Provided by me
- ▶ We will use tools to automatically detect common code
  - ▶ **All groups** with common code will be penalized
  - ▶ You cannot show or share code

The lab assignments are identical to those of last year, but this is no excuse

- ▶ The penalty may range:
  - From a zero in that lab **and** a penalty of “2 valores” in your final grade;
  - To failing the course (especially, in case of recidivism or of the project)
- ▶ The delay in publishing the grades is no excuse.



# Important Dates

## Labs

Lab	First lab class	Topic
Lab 0	23-09	Redmine, Devel. and SVN
Lab 2	30-09	Timer
Lab 3	14-10	Keyboard
Lab 4	28-10	Mouse
Lab 5	11-11	Video (graphics)

- Submission deadline of each lab is at the end of the lab's last lab class

## Programming Test

- Week starting on the 2019-11-25

## Project

What	Week	Comments
Proposal	11-11	7th lab class
Specification	18-11	8th lab class
First demo	16-12	Last lab class
Submission	2019-01-06 @ 20:00	all
Presentation	8, 9 and 10 January, 2020	To be confirmed

# Lab Sections

- ▶ I've instructed the secretariat not to accept more than 24 students per section
  - ▶ ensure you get help from staff, if you need it
  - ▶ each group should have only 2 students
- ▶ Any changes will have to be done by permutation among sections
  - ▶ We have created a forum in LCOM's Moodle explicitly with that purpose
- ▶ **In the lab classes, students will be allowed to join only groups in their sections**

# Announcements

## Lectures

- ▶ Start 10 minutes after the hour, e.g. 15:10 on Mondays
  - ▶ Actually, this is a FEUP's rule, and therefore applies also to lab classes

## Labs

- ▶ Start next Monday, i.e. September 23rd
  - ▶ All sections will have 12 lab classes

# Acknowledgments

- ▶ Prof. António Miguel Pimenta Monteiro (who designed the course)
- ▶ Prof. João Cardoso (not the same person as TCOM's lecturer) (who perfected it)
- ▶ Prof. Pedro Silva, who made possible last year's changes:
  - ▶ has proposed, designed and implemented the LCOM Framework;
  - ▶ implemented a set of utilities that make the development process in LCOM easier;
  - ▶ ported libx86emu to Minix, allowing us to use the most recent Minix version, and thus use:
    - ▶ more recent compiler with better error/warning messages;
    - ▶ VirtualBox shared foldersall of which also simplify the development process in LCOM
- ▶ developed a set of scripts to automatically generate a new VirtualBox image with Minix 3.4.0rc6

# Thank You!

# Questions?

# Platform

## MINIX 3

Unix-like operating system that allows privileged user processes to:

- ▶ Access every memory address
- ▶ Access directly I/O devices
- ▶ Process interrupts

## Linux

- ▶ MINIX 3 is installed in a VirtualBox VM

# Software

- ▶ CLANG compiler and assembler
  - ▶ This is Minix's default compiler and provides much better messages than those of the GNU compiler available in Minix 3.1.8
- ▶ Other SW development tools
  - ▶ make
  - ▶ SVN
  - ▶ doxygen
  - ▶ ar
- ▶ Redmine



# Advice for Course Participation

- ▶ It is important that you participate in the lectures
  - ▶ Especially this year, that each lecture is 2 hours
- ▶ It is also important that you prepare for the lab classes
  - ▶ Unless you prepare your labs, it is unlikely you will be able to complete them during the respective class
- ▶ To motivate you to do both:
  - ▶ In the 2nd hour of each lecture, we will present the main material required for the lab class of the following week.  
E.g., today, we will present the material for next week's lab: Lab 0
  - ▶ In the first hour of each lecture, we will answer to questions about that material as well as about the lab work planned for that week
    - ▶ We will give priority to questions made through a Moodle forum that we will create.
    - ▶ These questions, will be used to assign you a grade to be used in the computation of course participation.
- ▶ It is important that you provide us the information we request you in a timely fashion.
  - ▶ Failure to do so, will affect your course participation score.