

## CmpE 322 – Operating Systems

### Project 1 – Exploring Minix and Implementing a Simple System Call

(Due Date: November 18<sup>th</sup>, 2016 – 23:59)

This document includes the details of your first project assignment. You need to have installed Minix OS, and downloaded the source code (details can be found in the PS slides), in order to accomplish the objectives given below. This project is an individual assignment, please do not attempt it as a group. It will make up 100/1000 points of your course grade.

- Please submit your project by sending an e-mail to [cantunca@gmail.com](mailto:cantunca@gmail.com).
- The subject of your email should be in the format “**CmpE322-Project1 Surname Name StudentID**”. Emails not following this subject format will not be considered!
- You need to prepare a **zip archive** containing the project files and include it as an attachment to your submission email. The details of the required project files are given at the end of this document.
- You are also required to include a project report in your project archive. The content and the format of the report are also given at the end.

#### Part 1: Modifying and Compiling the Minix Kernel (40/100 pts)

In this part of the project, you will explore the Minix source code, modify and recompile some parts of it to achieve the objectives below. In order to prevent system errors or file losses, please backup the original source files into a different directory (such as `src_orig`). You will also need the original source files to extract a patch file which will be included in your submission.

1. Go to the kernel source and find the part that contains banner information at the startup. Edit this file in order to print “This kernel is modified by <Surname, Name, StudentID> for the course CmpE322” at the startup. **(10 pts)**
2. Please find the file that contains `OS_VERSION` and change it to version “4.0”. **(10 pts)**
3. Please locate the file that is responsible for reading the boot image and process queue at the boot. **(20 pts)**
  - a. Modify the necessary file in order to print the name of each process involved in the boot process.
  - b. Modify the necessary file in order to print the total number of processes involved in the boot process.

#### Part 2: Implementing a Simple Minix System Call (50/100 pts)

In this part, you will implement a simple system call as part of the process manager (PM) server with the name **printinteger**. Similar to the previous part, the objective is to find where the necessary code should be placed, rather than writing too much code.

1. The system call **printinteger** should take an integer as an argument and print the following line:  
“System call PRINTINTEGER called with value <passed integer> (<Surname, Name, StudentID>).”
2. It should be callable by the syntax (similar to all other system calls):  
**`_syscall(PM_PROC_NR, PRINTINTEGER, &m)`**

3. You should also associate this system call with a user library function with the alias, `int printinteger(int value)`, so that it can be called anywhere from a C program.
4. Finally, you need to create a small C test program (placed anywhere you like in the OS filesystem) that calls your user library function. You must include this test program separately in your project zip archive, if it is placed in a directory other than `src`.

**Important:** You need to create a patch file and include it in your project zip archive to be submitted. Please do not submit each modified file separately (except for the test program that calls the user library function). The patch file can be extracted via the “diff” command:

- Diff command needs to be used in order to create a patch file which will be used on the clean Minix source files in order to get your modified version! Please refer to the PS materials for more information.
- The first directory should be your clean `src` directory (backed up version) and the second directory is your modified one. The patch needs to list the differences between these directories and should be applicable on a clean source directory.

### Report (10/100 pts)

You should write a short report with separate sections for each of the two parts, and answer the following questions. You should include this report in your project zip archive as well.

Part 1:

- Where exactly are the source files that you modified? List the directories, filenames and the brief locations of modifications within the files.
- Briefly discuss the modifications you did and what they exactly do.
- What steps did you take to recompile the kernel and boot with the modified version?

Part 2:

- Where exactly are the source files that you modified/created? List the directories, filenames and the brief locations of modifications within the files.
- Briefly discuss the modifications you did and what they exactly do.
- Provide instructions on how to compile and run your test program (that calls the user library function that you implemented).

In summary, your zip archive should contain the following items:

- The diff patch of the `src` directory, containing all the modifications and additions you did for parts 1 and 2.
- The test program file created in Part 2.
- Project Report in .pdf or .docx format.

**One final remark:** Installing Minix OS, and building its source code can be a difficult process, especially for the first time. Those who visit the PS sessions regularly should already have accomplished this; however, if you neglected to do so, you are advised to start early to avoid any last minute problems. You may refer to the PS slides and the troubleshooting emails sent by us through the semester, to resolve typical problems that are encountered during installation.