

Lab 5

- Log on **Linux**
- On *System tab*, open **virtualbox**
- *File - Import - Appliance*
- Open Appliance
- On *File System*: **/var/OVAs/Debian-6.0-rev4-2012.ova**
- Open
- Next
- Scroll to the end
- Change *path of the virtual disk image* to
- **/var/tmp/sXXXXXX/Debian 6.0 laf-2012-disk1.vmdk**

- Check/install **nasm**
- Check/install **ddd**
- Check / install the machine emulator and virtualizer **qemu**

- Read the file **Booting a PC to run a kernel.pdf**
- Read the file **Installing GRUB on a hard disk image file.pdf**
- Follows with the **ddd** debugger the steps from bootstrap to the **GRUB** kernel selection menu

- No report is necessary for this.

Lab 5.1

Follows the steps for creating the kernel and the bootable hard **hd.img** file in directory **MK/paging**

Use the tutorial in http://www.jamesmolloy.co.uk/tutorial_html/ to understand the main and the functions of this minimal kernel.

- Change the file **main.c** to verify which the first address that produces a page fault is.
Hint: Increment ptr , starting from 0, of the page dimension (take care of the pointer arithmetic).
Use the functions **monitor_write**, **monitor_write_dec**, **monitor_write_hex** to print lines such as:
Normal access at address 0x0 at page 0
Normal access at address 0x1000 at page 1
..

Take note of the **number of pages** that your kernel has allocated.

Lab 5.2

Write another file **main1.c** that

- Writes the page number on the first address of each page
- Prints the content of the content of the first address of all mapped pages (the number information is the one derived from previous main). Ex.:
prt 0x0 (page 0) contains 0
prt 0x1000 (page 1) contains 1
- Swap frame 0 and 1 in the page table
- Prints the content of the first address of these pages, so that you should obtain an output such as:
page 0 now contains 1
page 1 now contains 0

Produce a report that illustrates the data structure of the directory page table and page table, and the functions that you have used.

Use figures and tables to make your report easily readable.

The delivery of the laboratory assignment must be performed using the link "**Elaborati**" in the course official page. You can upload your file logging to your page in the "didattica" site.

Create a compressed **tgz** archive file with this syntax:

Lab <LAB_number>_<Your_Id_number>_<YourLastname>. tgz

Example:

Lab05_123456_Smith.tgz

which includes all the source programs, and if necessary the input data, for each assignment.

For example: **Lab5-123456_Smith.tgz** must include **your report, and the files main.c, main1.c** and other functions that you modified.

Drop your **tgz file** in the folder **Elaborati** of the SDP site of Portale della didattica.