

# Operating Systems – Laboratory

Hristo Trifonov

6/February/2019

*For SDip in Embedded Systems Engineering*

## Laboratory Assignment #2

### OBJECTIVES:

The objectives of this laboratory assignment are as follows:

- 1) Learn some of the basic bash shell commands
- 2) Learn how to write and run a very simple shell script program
- 3) Learn how to extract some key system specifications and report them in a script

### INSTRUCTIONS:

- Students will provide individual submissions (however, learning/study cooperation is encouraged).
- Students can be asked to demonstrate their solutions.
- A short report **document** file must be submitted to describe the operation of your program and to comment on any problems etc. (see Addendum for details).

### SUBMISSION:

Students will submit via SULIS (EE5012 page) by 23:55 hours, Wednesday 13<sup>th</sup> February 2019. Late reports will not be accepted. The submitted files will be the:

- **system\_info** plain script file
- a **report** file (pdf) – see format in the Addendum

Please put all of the above in a folder named “**Assignment2\_yourIdNumber**” and compress/archive with **zip, tar** or whichever program you prefer before submitting to SULIS.

The student name and ID number is to be on the heading comments of any script file. The programs are to be commented for readability. Individual submissions only will be accepted. The student can be asked to demonstrate the working programs in the lab.

### Assignment assessment weightings:

Assignment #1	10% of module
<b>Assignment #2</b>	<b>10% of module .... this assignment</b>
Assignment #3	10% of module
Assignment #4	10% of module

*There will be a compulsory exam question in the final exam based on the laboratory assignments.*

# INSTRUCTIONS

## STEPS:

- 1) Log on to your Linux system and become familiar with the desktop environment and in particular the terminal
- 2) Read/study EE5012 UNIT 2 document
- 3) Ensure that you understand the sample questions Q1, Q2, Q3 and Q4 in the EE5012 UNIT 2 Appendix document
- 4) Ensure that you can write simple script programs as in UNIT 2 Appendix document
- 5) Write the script program **system\_info** as detailed below

Note – you are being asked to write your first very simple script file; that does not use any complicated variables, does not use any simple arithmetic, and does not use any programming constructs such as if, while loops etc. Thus the emphasis is on getting started with script programming and getting familiar with the editor and terminal and running the scripts.

## NOTE:

Include comments in your script.

Include display text for the user for a neat informative display e.g.

**The total number of files is: 22**

---

## EXERCISE:

Write a shell script called **system\_info** to do the following.

Display/print the following information:

- The total **number of files** that are in your **home** directory (including the files in all sub-directories. *Hint: use the “find” command*) and the total **number of directory files** in your **home** directory. (*Hint: see UNIT 2 Appendix document – discussion on Q3 i.e. `dirNum=$(ls -l ~ | grep “^d” | wc -l)` etc.*)
- The number of **free blocks** and the **percentage of used space** on the partition where your **home directory** resides
- The name of the **user**, the **user ID**, the **host name** of the machine and the **machine type**. (*Hint: Use the relevant environment variables shown by the `set` command, for example to display the current shell use something like: `echo “The shell is: $SHELL”`)*)
- List the **CPU model name**, the **CPU speed**, the **CPU cache size**, the **CPU vendor ID**, and the **number of CPU cores**. (*Hint: in the `/proc` directory extract the information from the `cpuinfo` file, for example `grep “^cpu family” /proc/cpuinfo`.) [NB: If using a non-Linux based UNIX operating system, then suggest an equivalent solution]*)
- List the **total memory size**, the **size of the free memory**, and the **size of available memory**. (*Hint: in the `/proc` directory extract the information from the `meminfo` file.*) [If using a non-Linux based UNIX operating system, then suggest an equivalent solution]

# ADDENDUM

## The document file

The submission for this laboratory assignment will include a document. Note, the document does not at all need to be very long and wordy – but must be of good quality, to the standard of a small technical report, and presented as listed below:

- 1) The file will be submitted as a **PDF** file.
- 2) The document will have the following information and sections

### **Front page**

Title page with student name, ID, date, module code, assignment number (e.g. *Assignment #1*)

### **Requirements**

Briefly summarise the assignment requirements from the assignment instructions in the handout.

### **Description of solution**

Describe your solutions noting any special problems or issues.

### **Testing and results**

State how you tested your program and record any results, using **screenshots** to show actual outputs.

### **Statement of completion**

Briefly make a statement saying that you have completed all of the requirements, or summarise any aspects that you could not complete.

### **Source code**

Include your source code as part of this document. You will also submit a separate plain text source code file or files as stated on the cover page of this assignment.

**NOTE: A student can lose up to 30% of assignment marks for a bad report.**