Operating Systems – Laboratory

Hristo Trifonov

20/February/2019

For SDip in Embedded Systems Engineering

Laboratory Assignment #3

OBJECTIVES:

The objectives of this laboratory assignment are as follows:

1) Learn some of the basic bash shell commands
2) Learn how to write shell script program to manage processes and files
3) Learn how to write a simple Bash shell function
4) Learn how to use "signal" and "trap" commands

INSTRUCTIONS:

- Students will provide individual submissions (however, learning/study cooperation is encouraged).
- 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 27th February 2019. Late reports will not be accepted. The submitted files will be the:

- a **report** file (pdf) – see format in the Addendum
- three scripts (one for each exercise), **file_stat**, **load_reduce** and **signal_trap**

Please put all of the above in a folder named **"Assignment3_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
Assignment #2 10% of module
Assignment #3 10% of module …. this assignment
Assignment #4 10% of module
There will be a compulsory exam question in the final exam based on the laboratory assignments.

INSTRUCTIONS

Please complete the following three exercises.

EXERCISE 1: file_stat

Objectives:

- Learn how to examine basic disk volume information

INSTRUCTIONS:

From the EE5012 Unit 3 Appendix study the example script program:
"# Program to check disk space usage against a specific % limit"

You will now write a script that is a small variation on this example.

Write a simple script file program, called **file_stat**, to do the following:

Display the number of **available blocks** and the **percentage of used space** on your **home** partition volume. If the number of available blocks is less than 5000000 then echo a warning to the terminal, saying "Disk space is running low, **n%** is used", where n% shows actual percentage of used space, else, reassure the user that disk space is sufficient.

EXERCISE 2: load_reduce

Objectives:

- Learn how to monitor a process using **ps, top …**

Study the problems in EE5012 Unit 3 Appendix under the heading 'Example questions: process statistics using the ps command'.

```
#! /bin/bash
# Meaningless program called busy_wait to simulate some activity

while true; do
  (( x++ ))
done
```

Write a Bash script program, called **load_reduce**, which will do the following in the order listed:

1) Run the **busy_wait** program to provide an example busy process, by putting the following lines early in your script:

```
chmod   a+x  busy_wait
./busy_wait   &
sleep 1              # delay required to allow time for busy_wait to start
```

   [Hint: first make sure you put the busy_wait script into your local directory]

2) Display the **PID number** for the **busiest** process.
   *[Hint: use the **ps** command and sort on the %CPU column and then get the PID number. You will assign this PID value to a variable for use later in step 3 of the program. Note – there are other ways to find the busiest process using the **ps** command but any working solution is acceptable.]*

3) Kill the **busiest process** and display a message saying the name (command column) of which process was killed.

EXERCISE 3: signal_trap

Objectives:

- Learn how to use a **trap** for a signal
- Learn how to write a **function** in Bash

INSTRUCTIONS:

Write a Bash script program, called **signal_trap**, which is described below.

Consider the script program on page 6 of the **EE5012 Unit 3 Appendix**, referred to as **SAMPLE ANSWER TO ABOVE**, and you will write something similar but not the same.

Make the following changes to the given script program on page 6 of the EE5012 Unit 3 Appendix.

Run the **busy_wait** script instead of the **progB** script

Modify the **trap_function()** as follows:

- Display the CPU utilisation for the **busy_wait** script.
  [HINT – you could use something like usage=$(ps -p $! --format %cpu | tail -1)]

- Kill the **busy_wait** program and exit. *[HINT – don't forget about $! as used on page 6 example]*

Test you program by hitting CTRL+C on your keyboard to see that your loop will be interrupted and your **trap function** will do what you expect.

NOTE: run you program using **bash signal_trap** as this will give a separate shell and will be better at catching the CTRL+C signal, as compared to using **./signal_trap** method.

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.