February 13th, 2008

LIACS, Operating Systems course, Spring Semester 2008 by Nies Huijsmans
Url http://www.liacs.nl/~shenstra/os

Lab Assignment #1, original author is dr. A.H. Deutz

Student assistant, Fabrice Colas (LIACS, room 133, fcolas@liacs.nl, 071 527 7033)

Reports to be sent by February the 27th at fcolas@liacs.nl with the following deliverables

1) http://www.liacs.nl/~shenstra/os/documents/assignment1_report.doc with your answers to the questions
2) C program **with comments** of Part A iv), **compiling** and **running**
3) C program **with comments** of Part B, **compiling** and **running**
4) your answer to Part D
5) in assignment1_report.doc, explain what you have done and learned from this lab? Finally, tell what your lab partner and eventually your colleagues taught you.

Check that each file contains
the name of the students, lab assignment #1, due date, and date turned in.
Pack and compress everything into a ZIP archive

Grading of the lab assignment

1) pay a very special attention on the documentation of your source code
2) eventually, refer to webpages, books or colleagues that you got ideas from
3) your final source code should **run** and **compile**

4) **optionally** report the portability issues when you tried to compile and run your program under other OS like Windows, MacOSX, Minix, other Linux OS

Student 1: Rick van der Zwet, 0433373

Student 2: <none>

The assignment runs tested on

- private machines
- Mac OS X 10.4.11 - i686-apple-darwin8-gcc-4.0.1 (GCC) 4.0.1 (Apple Computer, Inc. build 5250)
- Linux 2.6.15-26 - gcc (GCC) 4.1.3 20070929 (prerelease) (Ubuntu 4.1.2-16ubuntu2)

Listing of the enclosed files (aka README):

Part	File	Description	How to compile and use? (which exercise)
	Makefile	Build all execacutables below	`make all #build all` `make clean #delete all binary`
A iv)	parse-arguments.c	Print arguments given	./parse-arguments arg1 arg2 arg3
B	simple-shell.c	Basic shell, with functions to execute a program and exit to perminate	./simple-shell

Part C: What is the PCB called in Linux?

The Process Control Block (PCB) is called Task Structure and Process Table reading http://en.wikipedia.org/wiki/Process_control_blockand http://www.faqs.org/docs/kernel_2_4/lki-2.html

Part D: Why did the designers of a shell use the sys calls fork and exec? What are the advantages of such a design?

By using fork and exec the where able to run multiple processes at ones and by using a new sub to call the user arguments the code of the shell is made more robust and less to be affected by malfunctioning programs

Additional question 1: What you have done in the lab?

- Asked a number of questions
- Wrote code part A,B

At home:

- Finished writing code
- Wrote report

Additional question 2: What you have done and learned from this lab? Finally, tell what your lab partner and eventually your colleagues taught you.

During the lab session I asked a number questions about the reason of using .doc report format, which is done cause having a uniform input will speedup the work needed to be done by the corrector. Next I have learned that writing a shell is not that trivial as it looks like. Even the most simple basic string operations takes a lot of time to program at C.