TopGear Assignments

Linux System Programming – L1

Assignments

Estimated Effort: 2 PDs

Total Points: 100

Required VDI: Linux

Author: Jayapathi Ramamohan

jayapathi.ramamohan@wipro.com

Date: 3-May-2018

This set of assignments tests application of following concepts of Linux System Programming

- File I/O [Topic No. 2 of LSP-L1]
- Process Management [Topic No. 3 of LSP-L1]
- Proper error reporting & exit status

To be considered as correct submission, code should meet the following guidelines:

- Source code filename : as specified in each exercise
- Input filename & output filename should not be hardcoded in program, unless it is stated as part of the problem specification
- File I/O operations should be done using system calls only, not using C library functions.
- Output should be as per the sample shown in each exercise.

Exercise – 1: Filename: fd.c

[ToC Mapping: ToC no. LSP-L1: Topics 2,3]

Write a Linux program that either proves or disproves the following:

- If a child process inherits a file descriptor from its parent process, then both the parent and child processes share the file descriptor.
    - Read/write operations performed on the file descriptor by either parent or child process impacts the read/write operations by the other process.

Source file must include description of the approach in comments and how to run the program.

Exercise – 2: Filename: ff.c

[ToC Mapping: ToC no. LSP-L1: Topics 2]

Write a program that finds all files having same inode number in a directory or its subdirectories.

Program should support the following requirements.

Requirement-1:

For the below command usage, list all files in the specified directory and its subdirectories, which have the same inode number as that of the filename specified with **–f** option.

$ ./ff **-f** filename directoryname

Requirement-2:

For the below command usage, list all files in the specified directory and its subdirectories, which have the same inode number as that of the inode number specified with **–i** option.

$ ./ff **-i** inodenumber directoryname

Requirement-3:

a) If files are found with the desired inode number, program should return with exit status 0.

b) If no files are found with the desired inode number, program should not print any output and return with exit status 1.

c) If filename and/or directory name specified on commandline does not exist, program should print appropriate error message to the standard error and should return with exit status 1.

d) If option specified is not a valid option, program should print appropriate error message to the standard error and should return with exit status 1.

--- END ---