ADVANCED OPERATING SYSTEMS

Course Syllabus

June 2014

Prerequisites: Operating Systems, C Programming

UNIT - I: PROCESSES AND SCHEDULING

Process States and System Call Interface; Life Cycle of a Process: Process Dynamics; Scheduler: working and implementation; Linux Process States and System Calls; Process Groups, Sessions, Foreground and Background Processes.

UNIT - II: INTERPROCESS COMMUNICATION AND SYNCHRONISATION

Signals, Pipes and Named Pipes (FIFOs); Threads and pthread library; Mutexes and Condition Variables; Semaphores; Producer-Consumer Problem and Solutions using mutexes, condition variables and semaphores.

UNIT - III: FILES AND FILE SYSTEMS

File and File Meta-data; File Naming Systems; File System Operations; File System Implementation; File System Structures; Booting an OS; File System Optimisation.

UNIT - IV: DEVICES AND DEVICE DRIVERS

Devices and Types of Devices; Terminal, Disk, SCSI, Tape and CD devices; Unification of Files and Devices; Device Drivers: Concepts and Implementation Details.

UNIT - V: RESOURCE MANAGEMENT AND SECURITY

Resource Management Issues; Types of Resources; Integrated Resource Scheduling; Queuing Models of Scheduling; Protection of Resources – hardware, software, and attacks; Security Policies.

SUGGESTED ASSIGNMENTS

1. Short programs with `fork()` and `exec()` family of system calls to create parent and child processes; impact on local, extern and static variables.
2. Study output of `ps` command in Linux and draw process trees, identify process groups, session and group leaders, foreground and backgound processes.
3. Write programs to explore the variety of signals and their behaviours.
4. Write programs for interprocess communication with pipes and `popen()` calls; pipes across parent and child processes; pipes across threads.

5. Producer-Consumer problem: restricted and general versions; solutions using mutexes, condition variables and semaphores.

6. Exploring `FILE` structure in Linux and the `fcntl()` and `ioctl()` calls.

7. Writing a simple device driver for a parallel port; extension to USB port (optional).

TEXTBOOKS:

Recommended:

- Charles Crowley. *Operating Systems: A Design-Oriented Approach*, Tata McGraw-Hill (2001 or later)
- Richard Stevens, Stephen Rago. *Advanced Programming in the Unix Environment*, Addison-Wesley (2013). Available for free download in PDF.

References:

- Maekawa, M. and Arthur E. Oldehoeft and Oldehoeft, R.R. *Operating Systems: Advanced Concepts*, Benjamin Cummings (1987). Available through Google Books.
- David A. Rusling. *The Linux Kernel*, http://www.tldp.org/LDP/tlk/tlk.html