Course Content

Linux Kernel Driver

- Introduction of Linux Device Drivers
  - The role of the device driver
  - Splitting the kernel
  - Classes of devices and modules
  - Joining the kernel development community
- Building and running modules
  - Setting up your test system
  - The Hello World Module
  - Kernel modules versus applications
  - Compiling and Loading
  - The kernel symbol table
  - Initialization and shutdown
- Char drivers
  - Major and minor numbers
  - Some important data structures
  - Char devices registration
  - open and release
  - scullâ€™s Memory Usage
  - read and write
  - Playing with the new devices

- Debugging Techniques
    - Debugging support in the kernel
    - Debugging by Printing
    - Debugging by querying
    - Debugging by Watching
    - Debugging System Faults
    - Debugging and related tools
- Concurrency and race conditions
    - Pitfalls in scull
    - Concurrency and its management
    - Semaphores and mutexes
    - Completions
    - Spinlocks
    - Locking Traps
- Advanced char driver operations
- Time, Delays,and Deferred work
    - Measuring Time Lapses
    - Knowing the Current Time
    - Delaying Execution
    - Kernel Timers
    - Tasklets
- Allocating memory
    - The Real Story of kmalloc

    - Get_free_page and Friends
    - vmalloc and Friends
    - Per-CPU Variables
    - Obtaining Large Buffers

- Communicating with hardware
    - Input/Output Ports and Input/Output Memory
    - Using I/O Ports
    - An I/O Port Example
    - Using I/O Memory
-

- Interrupt Handling
    - Preparing the Parallel Port
    - Installing an Interrupt Handler
    - Top and Bottom Halves
    - Interrupt Sharing
    - Implementing a Handler

- Data types in kernel
    - Use of Standard C Types
    - Assigning an Explicit Size to Data Items
    - Interface-Specific Types
    - Other Portability Issues
    - Linked Lists

- PCI drivers

    - The PCI Interface
    - A Look Back: ISA
    - PC/104 and PC/104+
    - Other PC Buses
    - SBus
    - NuBus

- USB Driver
    - USB Device Basics
    - USB and Sysfs
    - Writing a USB Driver
    - USB Transfers Without Urbs
    - Concept related to USB drivers

- The linux device model
    - Kobjects, Ksets, and Subsystems
    - Low-Level Sysfs Operations
    - Hotplug Event Generation
    - Buses, Devices, and Drivers
    - Putting It All Together
    - Hotplug
    - Dealing with Firmware

- Memory mapping and DMA
    - Memory Management in Linux
    - The mmap Device Operation

    - Performing Direct Input/Output
    - Direct Memory Access
- Network Drivers
    - How snull Is Designed
    - Connecting to the Kernel
    - The net_device Structure in Detail
    - Opening and Closing
    - Packet Transmission
    - Packet Reception
    - Receive Interrupt Mitigation
    - Changes in Link State
    - The Socket Buffers
- TTY driver
    - A Small TTY Driver
    - tty_driver Function Pointers
    - TTY Line Settings
    - ioctls
    - proc and sysfs Handling of TTY Devices
    - The tty_driver Structure in Detail
    - The tty_operations Structure in Detail
    - The tty_struct Structure in Detail
-