**Operating Systems Principles**

**Overview**

|  |  |
| --- | --- |
| Level | 2 (Semester 4) |
| Duration | 4 weeks |
| Lectures | 10x40 minutes/week for 2 weeks |
| Practicals/tutorials | 5x2 hours/week for 3 weeks |

**Learning Outcomes**

* Students will demonstrate a general knowledge and understanding of core principles and concepts of modern operating systems in terms of architecture, and in terms of abstractions such as processes, memory management, and storage management.
* Students will demonstrate a good understanding and the ability to analyse advantages and disadvantages (performance, security) of process and thread management (user vs. kernel mode), scheduling algorithms, issues of concurrency and resource competition (race conditions), virtual memory management, and efficient file system and I/O management.
* Students will demonstrate competence in recognizing and applying operating system features in the design and implementation of applications (e.g. thread programming, process communication, managing concurrency through semaphores and monitors), and assess and communicate effectively performance and security issues at an appropriate level of technical depths.

**Syllabus**

1. Course Introduction
2. Computer Architecture, Interrupt handling, Interrupts, Mode Switch, Memory Hierarchy, Operating System Management Tasks
3. Operating System Architecture, System Calls, Modes of Operation and Mode Switch, Kernel Architectures
4. Processes and Threads, Process Execution, Context Switch
5. Processes and Threads, Process Creation in Unix (fork / exec)
6. Processes and Threads, Multithreading, Thread Management in Unix, Thread Execution, User / Kernel Threads, Thread Programming
7. Memory Management, Logical / Physical Addressing Memory, Internal / External Fragmentation
8. Memory Management, Virtual Memory Management, Paging, Segmentation
9. Memory Management, Page Table Management (Hierarchical, Inverted), TLB Translation, Principle of Locality
10. Memory Management, Page Replacement Strategies, Clock Algorithm, Resident Set, Working Set
11. File Management, File Systems, File Allocation Strategies, File Allocation Table
12. File Management, Indexed Allocation (hierarchical Index), Free Space Management, Recovery, Journaling File Systems, File Access Control
13. Scheduling, Preemptive / Non-preemptive Scheduling Algorithms (FCFS, RR)
14. Scheduling, Priority Scheduling, Multilevel Feedback Scheduling
15. Scheduling, Realtime Scheduling
16. Synchronisation, Concurrency, Race Condition, Mutual Exclusion, Condition Synchronisation, Critical Section Problem and Solutions
17. Synchronisation, Software and Hardware Solutions to the Critical Section Problem, Dekker’s Algorithm, Peterson’s Algorithm
18. Synchronisation, Semaphores
19. Synchronisation, Semaphore Applications (Producer – Consumer Problem, Ring Buffer, Dining Philosophers)
20. Synchronisation, Monitor (Mutual Exclusion, Condition Synchronisation)
21. Deadlocks, Deadlock Prevention, Deadlock Avoidance, Banker’s Algorithm, Deadlock Detection