CS - 403 (Operating Systems)

Second Semester

Course Description

Course Code Number	**CS-403**	**Course Title**	**Operating Systems**
Semester Hours	Total 4 hours per week Lecture 2 hours per week Lab2 hours per week	**No. of Credit Units**	3
Prerequisite	**CST-301**	**Course Coordinator**	Dr. Khine Moe Nwe Faculty of Computer Science
Course Length	15 Weeks	**Type of Instruction**	Lecture + Lab

Course Objective

This course intended to provide a thorough discussion of the fundamentals of operating system designs, and to relate these to contemporary design issues and to learn the current directions in the development of operating systems.

Course Outline

A state-of-the art survey of operating system principles that covers fundamental technology as well as contemporary design issues, such as concurrency, deadlock, virtual memory management, real-time systems and multiprocessor scheduling, embedded OSs, virtualization techniques.

Learning Outcomes

Computer Science students gain the knowledge of real-world design choices with case studies in Linux, UNIX, Android, and Windows 10. Students able to keep pace with a complex and rapidly changing field through the comprehensive coverage of the latest trends and developments in operating systems.

Text Book

[1] Operating Systems Internals and Design Principles (9th Edition) by William Stallings

Reference Books

[1] Operating Systems Internals and Design Principles (7th Edition) by William Stallings

[2] Operating System Concepts (6th Edition) by Avi Silberschatz , Peter Baer Galvin, and Greg Gagne

Course Organization

Student participation in this course will involve the following activities:

- Attending the lectures
- Lab
- Test (Moodle)
- Quiz
- Assignments
- Exam

Assessment Plan for the Course

Paper Exam	50 %
Assignment/Project	15 %
Quizzes/ Moodle	15 %
Class Participation	10 %
Lab Test	10 %

Tentative Lecture Plan (30 Periods for 15 weeks)

No.	Topics	Week	Remark
	Concurrency: Mutual Exclusion and Synchronization		**Chapter (5)**
1.	5.1. Mutual Exclusion: Software Approaches 5.2. Principles of Concurrency	Week 1	
2	5.3. Mutual Exclusion: Hardware Support 5.4. Semaphores	Week 2	
3	5.5. Monitors 5.6. Message Passing 5.7. Readers/Writers Problem	Week 3+4	
	Concurrency: Deadlock and Starvation		**Chapter (6)**
4.	6.1. Principles of Deadlock 6.2. Deadlock Prevention 6.3. Deadlock Avoidance	Week 5	
5.	6.4. Deadlock Detection 6.5. An Integrated Deadlock Strategy 6.6. Dining Philosophers Problem	Week 6	
6.	6.7. Unix Concurrency Mechanisms 6.8. Linux Kernel Concurrency Mechanisms 6.9. Solaris Thread Synchronization	Week 7	

	Primitives 6.10. Windows Concurrency Mechanisms 6.11. Android Interprocess Communication		
	Virtual Memory		**Chapter (8)**
7.	8.1. Hardware and Control Structures 8.2. Operating System Software	Week 8	
8.	8.3. Unix and Solaris Memory Management 8.4. Linux Memory Management 8.5. Windows Memory Management 8.6. Android Memory Management	Week 9	
	Multiprocessor, Multicore, and Real-Time Scheduling		**Chapter (10)**
9.	10.1. Multiprocessor and Multicore Scheduling 10.2. Real-Time Scheduling	Week 10	
10.	10.3. Linux Scheduling 10.4. Unix Svr4 Scheduling 10.5. Unix Freebsd Scheduling 10.6. Windows Scheduling	Week 11	
	Embedded Operating Systems		**Chapter (13)**
11.	13.1. Embedded Systems 13.2. Characteristics of Embedded Operating Systems 13.3. Embedded Linux 13.4. Tinyos	Week 12+13	
	Virtual Machines		**Chapter(14)**
12.	14.1. Virtual Machine Concepts 14.2. Hypervisors 14.3. Container Virtualization 14.4. Processor Issues 14.5. Memory Management 14.6. I/O Management 14.7. Vmware Esxi 14.8. Microsoft Hyper-V and Xen Variants 14.9. Java Vm	Week 14+15	

Tentative Plan for Lab (30 Periods for 15 weeks)

No	Chapter	Periods
I	**Chapter 6. Controlling Access to Files with Linux File System**	**4**
1	Linux File System Permissions Managing File System Permissions from the Command Line	
	Practice: Interpreting File and Directory Permissions Practice: Managing File Security from the Command	
2	Managing Default Permissions and File Access	
	Practice: Controlling New File Permissions and Ownership Lab: Controlling Access to Files with Linux File System Permissions	
II	**Chapter 7. Monitoring and Managing Linux Process**	**4**
3	Processes Controlling Jobs	
	Practice: Processes Practice: Background and Foreground Processes	
4	Killing Processes Monitoring Process Activity	
	Practice: Killing Processes Practice: Monitoring Process Activity Lab: Monitoring and Managing Linux Processes	
III	**Chapter 8. Controlling Services and Daemons**	**2**
5	Identifying Automatically Started System Processes Controlling System Services	
	Practice: Identify the Status of systemd Units Practice: Using systemctl to Manage Services Lab: Controlling Services and Daemons	
IV	**Chapter 9. Configuring and Securing OpenSSH Service**	**4**
6	Accessing the Remote Command Line with SSH Configuring SSH Key-based Authentication	
	Practice: Accessing the Remote Command Line Practice: Using SSH Key-based Authentication	
7	Customizing SSH Service Configuration	
	Practice: Restricting SSH Logins Lab: Configuring and Securing OpenSSH Service	
V	**Chapter 10. Analyzing and Storing Logs**	**4**
8	System Log Architecture Reviewing Syslog Files Reviewing systemd Journal Entries	
	Practice: System Logging Components Practice: Finding Log Entries Practice: Finding Events With journalctl	
9	Perserving the systemd Journal Maintaining Accurate Time	
	Practice: Configure a Persistent systemd Journal Practice: Adjusting System Time Lab: Analyzing and Storing Logs	
VI	**Chapter 11. Managing Red Hat Enterprise Linux Networking**	**4**
10	Networking Concepts Validating Network Configuration Configuring Networking with nmcli	

	Practice: Networking Concepts Practice: Examining Network Configuration Practice: Configuring Networking with nmcli	
11	Editing Network Configuration Files Configuring Host Names and Name Resolution	
	Practice: Editing Network Configuration Files Practice: Configuring Host Names and Name Resolution Lab: Managing Red Hat Enterprise Linux Networking	
VII	**Chapter 12. Archiving and Copying Files Between Systems**	**4**
12	Managing Compressed tar Archives Copying Files Between Systems Securely	
	Practice: Backing Up and Restoring Files From a tar Archive Practice: Copying Files Over the Network With scp	
13	Synchronizing Files Between Systems Securely	
	Practice: Synchronizing Two Directories Securely With rsync Lab: Archiving and Copying Files Between Systems	
VIII	**Chapter 13. Installing and Updating Software Packages**	**4**
14	RPM Software Packages and Yum Managing Software Updates with yum	
	Practice: RPM Software Packages Practice: Installing and Updating Software with yum	
15	Enabling yum Software Repositories Examining RPM Package Files	
	Practice: Enabling Software Repositories Practice: Working with RPM Package Files Lab: Installing and Updating Software Packages	