

Yarmouk University
Faculty of Information Technology and Computer Sciences
Department of Computer Sciences

CS 630: *Advanced Operating Systems*

Course Credits: *3 Cr. Hrs.* **Prerequisite(s):** *CS231 or equivalent*

Semester/Session: *Fall 2008/2009*

Class Section(s) Time & Location:

Section 1, Sunday, Tuesday 3:30: - 4:45, Al-Magdisi Rm. 203

Instructor: *Dr. Najib Kofahi* **E-mail address:** nkofahi@yu.edu.jo

Office Hours: *Sunday - Thursday 9:00 - 10:00*

Office Location: *Al-Magdisi Bldg. 313, Third Floor*

Course Brief Description:

Advanced concepts in operating systems, inter-process communication and synchronization, distributed operating systems (concurrent processes, design techniques), file system, memory management, deadlock, protection and security, case studies from Linux, MACH, XP.

This course builds on the foundation material presented in CS231: Introduction to Operating Systems. It addresses and explores internals of contemporary operating systems, regarding issues listed in the catalog description.

Topics generally covered in advanced operating systems may include reference to relevant developments in hardware, distributed systems, networking, multiprocessor systems, real-time systems, multimedia systems, security, performance evaluation and applications.

Keep in mind that, traditionally, advanced operating systems are divided into several categories including: Distributed, multiprocessor, database operating systems, Multimedia operating systems, real-time operating systems, etc. Each of these topics may require a complete semester or even more to cover all the details!!!

In this course, we also take a closer look at the directions of recent research.

Note that File system, memory management, and deadlock are covered in the undergraduate course so it is given as reading assignment.

Course Objectives

- Familiarize students with traditional and nontraditional operating systems
- Exploring the internals of traditional and nontraditional operating systems including UNIX, Linux, XP and some distributed operating systems through case studies.
- Pointing out issues of design, programming, and implementation of operating system components through reading research papers and performing programming projects.
- Provide students with advanced concepts in traditional and nontraditional operating systems including distributed operating systems.
- Develop intuition in the student for which approaches work, and which don't.
- To develop the ability to sense where bottlenecks lie in system design.

Learning Outcomes

After completing this course the student must demonstrate the knowledge and ability to:

- Use traditional and nontraditional operating systems
- Recognize the differences between various types of operating systems.
- Study and explore the internals of new operating system.
- Deal with issues of design, programming, and implementation of operating system components through reading research papers and performing programming projects.
- Understand advanced concepts in traditional and nontraditional operating systems.
- Know where to look for more information when he is faced with an OS problem.
- Sense where bottlenecks lie in system design.
- Learn how to be critical of what he is told by system designers.

Teaching Methods

The course will be based on the following teaching and learning activities:

- Lectures covering the theoretical part using word and PowerPoint presentations.
- Reading recent research papers in the areas of OS
- Individual or group projects
- Case Studies

Evaluation Plan

Students will be evaluated in this course using a combination of assessment methods, including:

- Written exams: 85 %
 - Exam 1 20 %
 - Exam 2 25 %
 - Final 40 %
- Project 15 %

Teaching Resources

Main Textbook

- *Operating System Concepts* by Siblingschatz, Galvin and Gagne. 6th Ed. 2003, John Wiley & Sons, inc.

Supplementary References

- *Modern Operating Systems Second Edition* by Andrew S. Tanenbaum, Prentice Hall, 2001.
- *Operating Systems* by William Stallings Prentice-Hall, Inc.
- *Distributed Operating Systems Concepts and Design*, 1997, by Pradeep K. Sinha, IEEE Computer Society Press.
- *Understanding the Linux Kernel, 2001*, D. Bovet and M. Cesati, O'Reily.
- *Advanced Programming in the UNIX Environment*, W. Richard Stevens. Addison-Wesley; 1992.
- **Electronic material placed on Yarmouk University site for this course:** Syllabus, exercises, announcements
- **Important Links**

Course Plan ([Tentative](#)):

Week 1

General Introduction

- Different system types (including Mainframe systems; Desktop systems; Multiprocessor systems; Distributed systems; Clustered systems; Real-time systems; Handheld systems).

- Feature Migration
- Computing Environments;

Week 2

Operating System Structures

- Simple Structure
- Layered Approach
- Microkernel System Structure
- Client/Server Structure
- Virtual Machines

Weeks 3 and 4

Processes and Threads

- Overview and Background
- Motivation
- Benefits
- User and Kernel Threads
- Single and Multithreaded Processes
- Multithreading Models:
 - Many-to-One Model,
 - One-to-One Model,
 - Many-to-Many Model;
- Threading Issues;
- Case Study (Pthreads, Solaris Threads, Windows Threads, Linux Threads)

Weeks 5 and 6

Process Synchronization

- Background
- Critical-Section Problem and different solutions
- Synchronization Hardware
- Semaphores
- Classical Synchronization Problems
- Critical regions
- Monitors;
- Case Study: Synchronization in Solaris 2 & Windows

Exam 1

Week 7 & 8

File Systems

- Directory structure and implementation;
- File system structure
- File system implementation
- Free space management
- Efficiency and performance
- Recovery (Consistency checking, Backup and Restore);
- Log-Structured file system;
- Case Study: UNIX and NFS.
- **RAID Structure**

Week 9

DISTRIBUTED SYSTEM STRUCTURES:

- Background
 - Advantages
 - Types of Distributed Systems
- Topology and Network Types
- Communication and Communication Protocols
- Design Issues

Weeks 10 & 11

DISTRIBUTED FILE SYSTEMS:

- Background
- Naming and transparency
 - Naming Structures
 - Naming Schemes
 - Implementation Techniques
- Remote file access
- Stateful versus stateless services
- File replication

Exam 2

Weeks 12 & 13

Security

- The Security Problem
- Scope of system security
- Computer and Network Security Requirements
- Authentication
- Types of Threats
- Computer System Assets
- Malicious Software
- Program threats
 - Trojan Horse, Trap Door, Stack and Buffer Overflow
- System Threats
 - Worms, Viruses, Denial of Service
- Security Systems and Facilities
- Intrusion Detection
- Management techniques to Improve System Security
- Encryption
- Computer Security Classifications
- Case Study: Windows NT

Weeks 14 & 15

Protection

- Introduction and Goals of Protection
- Domain of Protection
 - Domain Structure
 - Examples: UNIX, MULTICS
- Access Matrix
 - Use of Access Matrix
- Implementation of Access Matrix
 - Global Table
 - Access Lists for Objects
 - Capability Lists for Domains
 - A Lock-Key Mechanism
 - Comparison of these methods
- Revocation of Access Rights
- Language-Based Protection.

NOTE: If time permit a selected subject of the following topics will be covered

- **OPERATING SYSTEMS DESIGN & IMPLEMENTATION ISSUES**
- **MULTIPROCESSORS and THEIR SCHEDULING**
- **REAL-TIME SYSTEMS and REAL-TIME SCHEDULING**

Important Notes:

- No make up examinations will be given!!!
- Yarmouk university rules and regulations will be enforced
- Attendance will be taken at the beginning of the class.
- Any student in position of an excuse for officially authorized absence must present this excuse to the instructor no later than three days following his resumption of class attendance.

COURTESY

I will expect students to be courteous toward the instructor and their classmates throughout the duration of this course. Talking while someone else is speaking will not be tolerated. Furthermore, all mobiles must be turned off during class. In addition, students are expected to be in class on time. Late arrivals will disrupt the class session and will be considered as absence.

Best of luck!!!