CSE 421/521 - Operating Systems
Fall 2012

LECTURE - I
INTRODUCTION

Tevfik Koşar
University at Buffalo
August 28th, 2012

Contact Information

- Instructor: Prof. Tevfik Kosar
  - Office: 338J Davis Hall
  - Phone: 645-2323
  - Email: tkosar@buffalo.edu
  - Web: http://www.cse.buffalo.edu/~tkosar
  - Office hours: Wed 11:00am - noon, Thu 11:00am - noon
    (Or anytime by appointment)

- Teaching Assistants:
  - Sonali Batra <sonaliba@buffalo.edu>
  - Ying Yang <yyang25@buffalo.edu>
  - Weida Zhong <weidazho@buffalo.edu>

Recitations

- You need to attend one of the following recitations:
  - Tue 11:00am - 11:50am (Capen 260)
  - Wed 10:00am - 10:50am (Obrian 112)

- Recitations will include:
  - Clarification of some important course material
  - Solutions of some exercise questions
  - Project & HW guidance
  - Programming tips

Course Web Page

- Course web page:
  http://www.cse.buffalo.edu/faculty/tkosar/cse421-521/
  - All lecture notes will be available online
  - As well as homework assignments, projects and other
    important course information

<table>
<thead>
<tr>
<th>Date</th>
<th>Lect.</th>
<th>Title</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aug 16</td>
<td>1</td>
<td>Introduction</td>
<td>Read Ch 1</td>
</tr>
<tr>
<td>Aug 16</td>
<td>2</td>
<td>Operating System Struct</td>
<td></td>
</tr>
<tr>
<td>Aug 16</td>
<td>3</td>
<td>Processes</td>
<td></td>
</tr>
<tr>
<td>Aug 22</td>
<td>4</td>
<td>Threads</td>
<td></td>
</tr>
<tr>
<td>Aug 22</td>
<td>5</td>
<td>CPU Scheduling I</td>
<td></td>
</tr>
<tr>
<td>Aug 29</td>
<td>6</td>
<td>CPU Scheduling II</td>
<td></td>
</tr>
<tr>
<td>Aug 29</td>
<td>7</td>
<td>Process Synchronization I</td>
<td></td>
</tr>
<tr>
<td>Sep 5</td>
<td>8</td>
<td>Process Synchronization II</td>
<td></td>
</tr>
<tr>
<td>Sep 5</td>
<td>9</td>
<td>Deadlocks - I</td>
<td></td>
</tr>
<tr>
<td>Sep 12</td>
<td>10</td>
<td>Deadlocks - II</td>
<td></td>
</tr>
<tr>
<td>Sep 19</td>
<td>11</td>
<td>Memory - I</td>
<td></td>
</tr>
<tr>
<td>Sep 19</td>
<td>12</td>
<td>Memory - II</td>
<td></td>
</tr>
<tr>
<td>Oct 10</td>
<td>13</td>
<td>Midterm Exam</td>
<td>8:30am 10:00am</td>
</tr>
<tr>
<td>Oct 10</td>
<td>14</td>
<td>Midterm Discussion</td>
<td></td>
</tr>
<tr>
<td>Oct 15</td>
<td>15</td>
<td>Interfaces</td>
<td></td>
</tr>
<tr>
<td>Oct 15</td>
<td>16</td>
<td>Programming II</td>
<td></td>
</tr>
</tbody>
</table>

Textbook: Required

Recommended Supplementary Text

Advanced Programming in the UNIX Environment
Second Edition
W. Richard Stevens
Stephen A. Rago

Noval AGREST PROFESSIONAL COMPUTER SERIES

Brian W. Kernighan
Dennis M. Ritchie
Recommended Supplementary Text

Recommended Supplementary Text

Grade Components

- The end-of-semester grades will be composed of:
  - Pop Quizzes : 5% (4-5)
  - Homework  : 10% (4)
  - Projects   : 30% (3)
  - Midterm   : 25% (1)
  - Final     : 30% (1)

* You are expected to attend the classes and actively contribute via asking and/or answering questions.

Grading Scale

- Final grades will be given according to this scale:

<table>
<thead>
<tr>
<th>Point Range</th>
<th>Letter Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>95.00-100</td>
<td>A</td>
</tr>
<tr>
<td>90.00-94.99</td>
<td>A-</td>
</tr>
<tr>
<td>85.00-89.99</td>
<td>B+</td>
</tr>
<tr>
<td>80.00-84.99</td>
<td>B</td>
</tr>
<tr>
<td>75.00-79.99</td>
<td>B-</td>
</tr>
<tr>
<td>70.00-74.99</td>
<td>C+</td>
</tr>
<tr>
<td>65.00-69.99</td>
<td>C</td>
</tr>
<tr>
<td>60.00-64.99</td>
<td>C-</td>
</tr>
<tr>
<td>55.00-59.99</td>
<td>D+</td>
</tr>
<tr>
<td>50.00-54.99</td>
<td>D</td>
</tr>
<tr>
<td>49.99-0</td>
<td>F</td>
</tr>
</tbody>
</table>

* I will use “curve” to adjust grades (up) to this scale.
* There will be separate curves for graduate & undergraduate students.

Rules

- No use of laptops/phones during the lectures!
- No late homework/project submissions accepted!
- Exams will be closed book.
- You are only responsible from material covered in the class, homework, and projects.
- Academic dishonesty will be treated “very” seriously!

Passive vs Active Learning

Passive learning: learning through reading, hearing & seeing
Active learning: learning through saying and doing

After 2 weeks, we tend to remember:

Passive learning
- 10% of what we read
- 20% of what we hear
- 30% of what we see (i.e. pictures)
- 50% of what we hear and see

Active learning
- 70% of what we say
- 90% of what we say and do

How to Become an Active Learner

- Recall prior materials
- Answer a question
- Guess the solution first (even guessing wrong will help you to remember the right approach)
- Work out the next step before you have to read on
- Think of an application
- Imagine that you were the professor and think about how you would give a test on the subject material so that key concepts and results will be checked.
- Summarize a lecture, a set of homework or a lab in your own words concisely.
What Expect to Learn?

- Key Concepts of Operating Systems
  - Design, Implementation, and Optimization
- Topics will include:
  - Processes, Threads and Concurrency
  - CPU and I/O Scheduling
  - Memory and Storage Management
  - File System Structures
  - Synchronization and Deadlocks
  - Protection and Security
  - Distributed Computing & Related Issues

What is an Operating System?

- A program that manages the computer hardware.
- An intermediary between the computer user and the computer hardware.
- Manages hardware and software resources of a computer.

Introduction

Computer System Overview

1. Hardware
   ✓ provides basic computing resources
   ✓ CPU, memory, disk, other I/O devices
2. Firmware (BIOS)
   ✓ software permanently stored on chip (but upgradable)
   ✓ loads the operating system during boot
3. Operating system
   ✓ controls and coordinates the use of the hardware among the various application programs for the various users
4. System programs
   ✓ basic development tools (shells, compilers, editors, etc.)
   ✓ not strictly part of the core of the operating system
5. Application programs
   ✓ define the logic in which the system resources are used to solve the computing problems of the users
   ✓ database systems, video games, business programs, etc.
6. Users
   ✓ people, other computers, machines, etc.
Role of an Operating System

- The Silberschatz “pyramid” view

- The Tanenbaum “layered” view

- The Stallings “layered & stairs” view

- The Molay “aquarium” view

Key Point

- An operating system is a program that acts as an intermediary between users/applications and the computer hardware.

Operating System Goals

- From the user perspective:
  - Executes user programs and make solving user problems easier
  - Makes the computer system convenient to use
    - hides the messy details which must be performed
    - presents user with a virtual machine easier to use

- From the System/HW Perspective:
  - Manages the resources
  - Uses the computer hardware in an efficient manner
    - time sharing: each program gets some time to use a resource
    - resource sharing: each program gets a portion of a resource
OS Services for Users

• Program Execution
  - The OS loads programs and data into memory, initializes I/O devices and files, schedules the execution of programs

• Access to I/O Devices
  - The OS hides I/O device details from applications (direct I/O access is forbidden) and offers a simplified I/O interface

• Controlled Access to Files & Directories
  - The OS organizes data into files and directories, controls access to them (i.e. create, delete, read, write) and preserves their integrity

OS Services for System/HW

• Resource Allocation
  - The OS allocates resources to multiple users and multiple jobs running at the same time

• Operation Control
  - The OS controls the execution of user programs and operations of I/O devices

• System Access
  - The OS ensures that all access to resources is protected, including authorization, conflict resolution etc.

• Accounting and Usage Statistics
  - The OS keeps performance monitoring data

Summary

• What is an OS?
• Role of an OS
• Operating System Goals
  - User View vs System View
• Operating System Services
  - For Users and HW

OS Services for Users

• Communications
  - The OS allows exchange of information between processes, which are possibly executing on different computers

• Error Detection and Response
  - The OS properly handles HW failures and SW errors with the least impact to running applications (i.e. terminating, retrying, or reporting)

Questions?

Acknowledgements

• “Operating Systems Concepts” book and supplementary material by A. Silberschatz, P. Galvin and G. Gagne

• “Operating Systems: Internals and Design Principles” book and supplementary material by W. Stallings

• “Modern Operating Systems” book and supplementary material by A. Tanenbaum

• R. Doursat and M. Yuksel from UNR

Reading Assignment: Chapter 1 from Silberschatz.