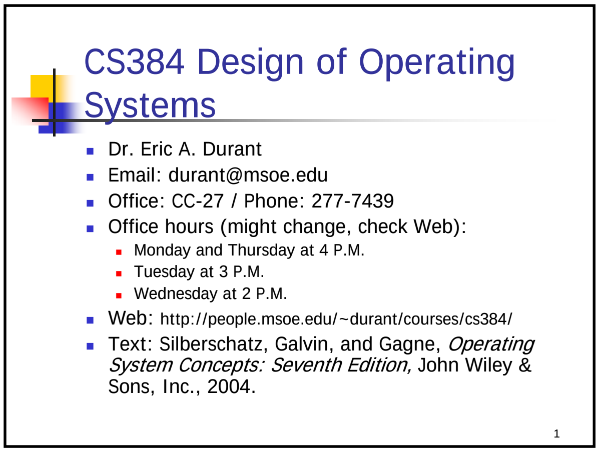
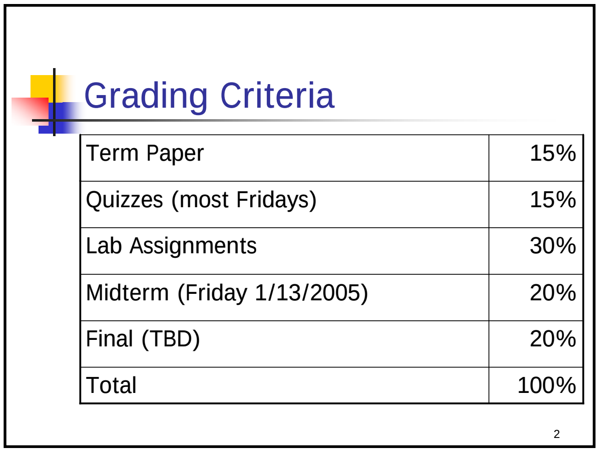


CS384 Design of Operating Systems

- Dr. Eric A. Durant
- Email: durant@msoe.edu
- Office: CC-27 / Phone: 277-7439
- Office hours (might change, check Web):
  - Monday and Thursday at 4 P.M.
  - Tuesday at 3 P.M.
  - Wednesday at 2 P.M.
- Web: http://people.msoe.edu/~durant/courses/cs384/
- Text: Silberschatz, Galvin, and Gagne, *Operating System Concepts: Seventh Edition,* John Wiley & Sons, Inc., 2004.

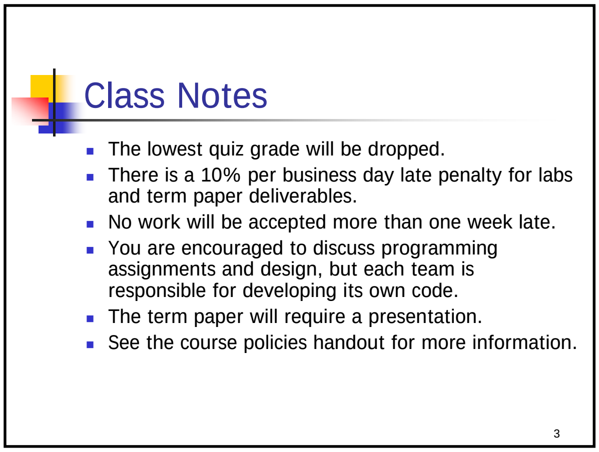
1



Grading Criteria

Term Paper	15%
Quizzes (most Fridays)	15%
Lab Assignments	30%
Midterm (Friday 1/13/2005)	20%
Final (TBD)	20%
Total	100%

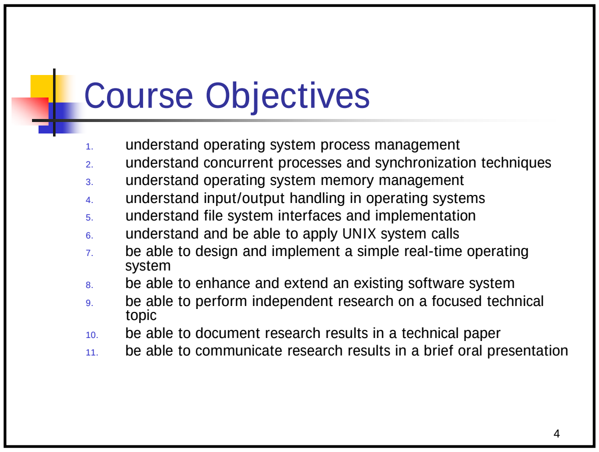
2



Class Notes

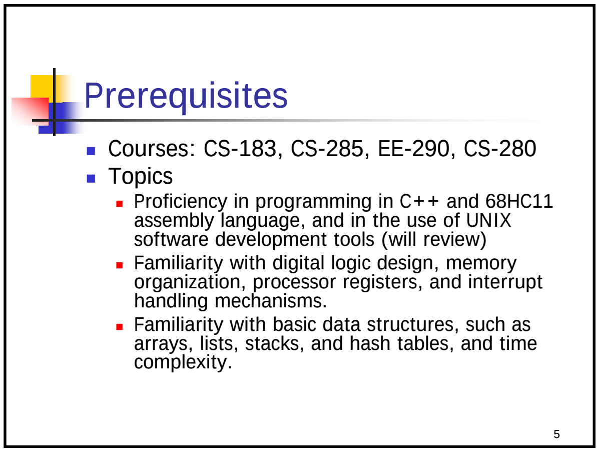
- The lowest quiz grade will be dropped.
- There is a 10% per business day late penalty for labs and term paper deliverables.
- No work will be accepted more than one week late.
- You are encouraged to discuss programming assignments and design, but each team is responsible for developing its own code.
- The term paper will require a presentation.
- See the course policies handout for more information.

3



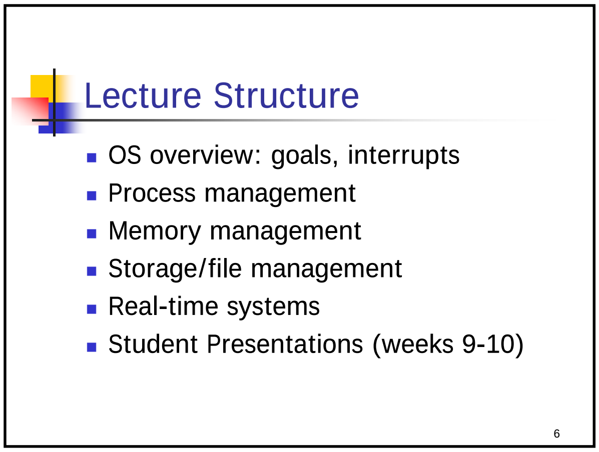
Course Objectives

1. understand operating system process management
2. understand concurrent processes and synchronization techniques
3. understand operating system memory management
4. understand input/output handling in operating systems
5. understand file system interfaces and implementation
6. understand and be able to apply UNIX system calls
7. be able to design and implement a simple real-time operating system
8. be able to enhance and extend an existing software system
9. be able to perform independent research on a focused technical topic
10. be able to document research results in a technical paper
11. be able to communicate research results in a brief oral presentation



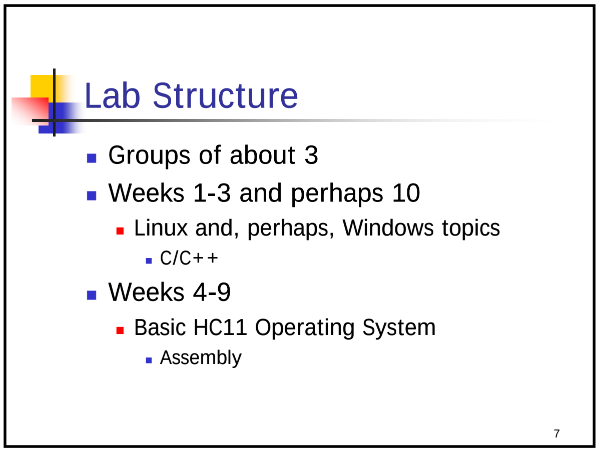
Prerequisites

- Courses: CS-183, CS-285, EE-290, CS-280
- Topics
  - Proficiency in programming in C++ and 68HC11 assembly language, and in the use of UNIX software development tools (will review)
  - Familiarity with digital logic design, memory organization, processor registers, and interrupt handling mechanisms.
  - Familiarity with basic data structures, such as arrays, lists, stacks, and hash tables, and time complexity.



Lecture Structure

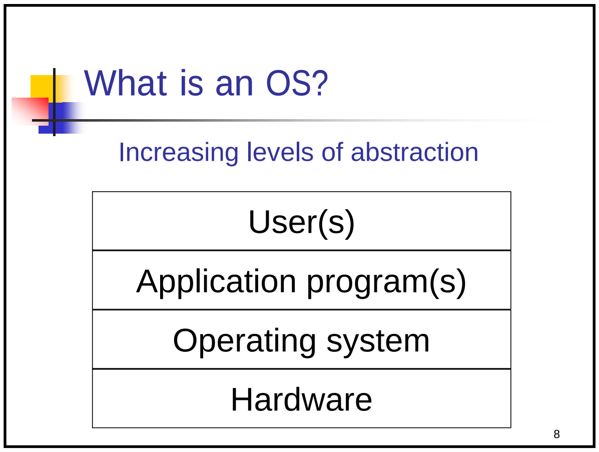
- OS overview: goals, interrupts
- Process management
- Memory management
- Storage/file management
- Real-time systems
- Student Presentations (weeks 9-10)



Lab Structure

- Groups of about 3
- Weeks 1-3 and perhaps 10
  - Linux and, perhaps, Windows topics
    - C/C++
- Weeks 4-9
  - Basic HC11 Operating System
    - Assembly

7



What is an OS?

Increasing levels of abstraction

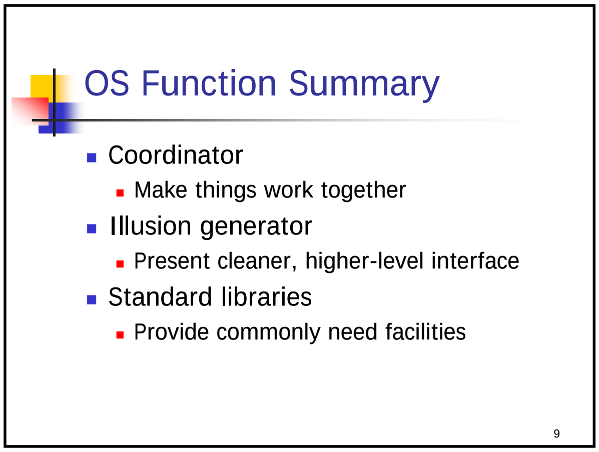
User(s)

Application program(s)

Operating system

Hardware

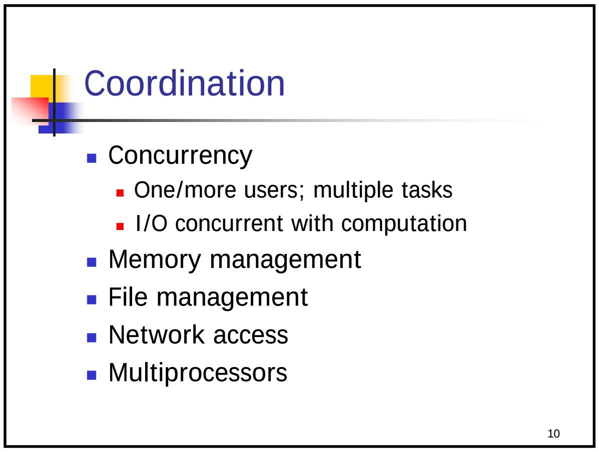
8



OS Function Summary

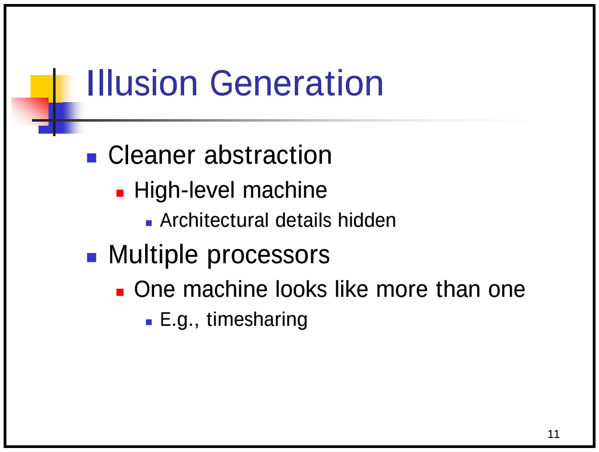
- Coordinator
  - Make things work together
- Illusion generator
  - Present cleaner, higher-level interface
- Standard libraries
  - Provide commonly need facilities

9



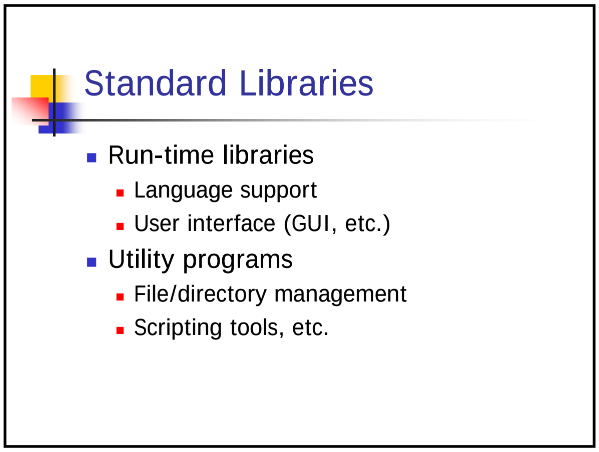
Coordination

- Concurrency
  - One/more users; multiple tasks
  - I/O concurrent with computation
- Memory management
- File management
- Network access
- Multiprocessors



Illusion Generation

- Cleaner abstraction
  - High-level machine
    - Architectural details hidden
- Multiple processors
  - One machine looks like more than one
    - E.g., timesharing



Standard Libraries

- Run-time libraries
  - Language support
  - User interface (GUI, etc.)
- Utility programs
  - File/directory management
  - Scripting tools, etc.