Operating Systems

CS2006

Lecture 1

Introduction

23rd January 2023

Myself

Dr. Rana Asif Rehman

Associate Professor

Email: **r.asif@lhr.nu.edu.pk**

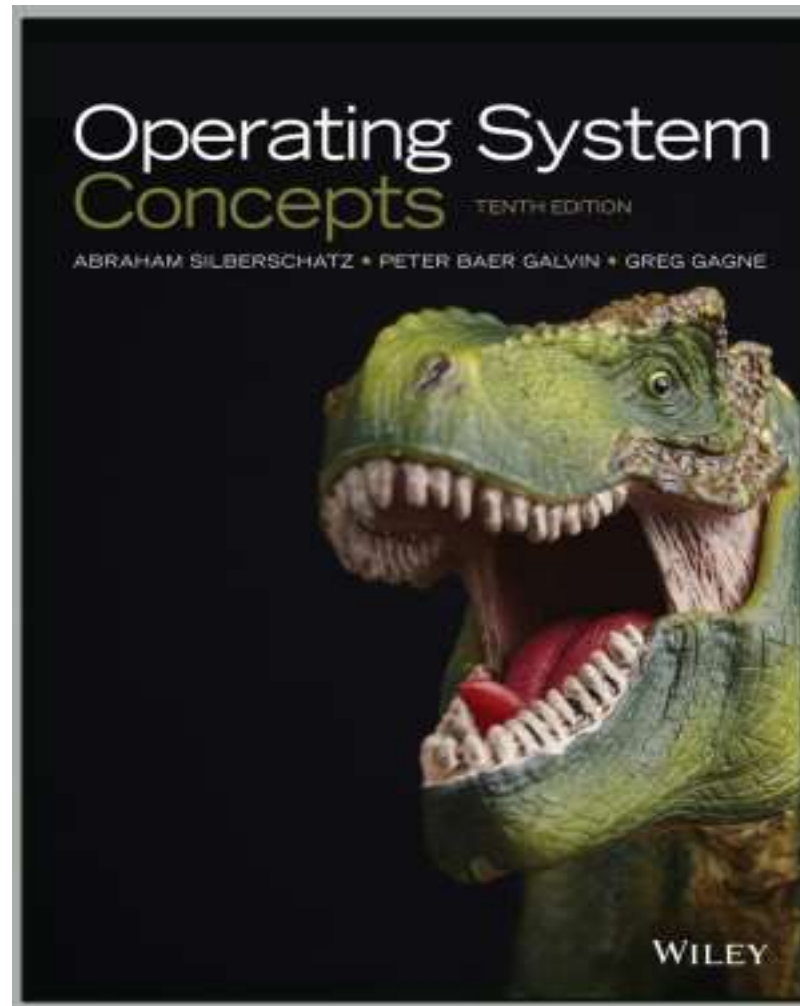Phone: (042) 111-128 -128   Ext: 455

Office: 1st floor, Library Building,
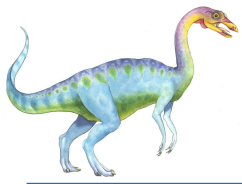
FAST School of Computing, FAST-NUCES, Lahore.

Books and Resources

- **Operating System Concepts** 10th Edition by Silberschatz, Galvin and Gagne

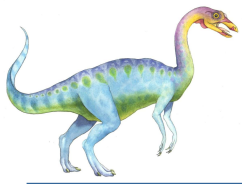# Google Classroom

☐ Please join the Operating Systems (**Section 4B**) class by using the following code:

<div align="center">

eioop4p

</div>

Course outline

- Introduction
- Processes, Inter-process communication,
- Threads
- Synchronization, Semaphores,
- Deadlocks
- CPU Scheduling
- Memory management
- File-system management
- ……

Class Rules & Regulations

- **Quizzes**
    - All quizzes will be **Announced**
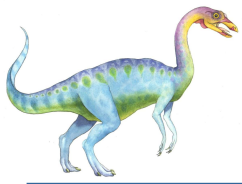   - Quizzes will have more weightage as compared to Assignments
- **Assignments**
    - In case of copied, all remaining assignment will be marked **Zero**
    - Avoid late submission, **No** assignment will be entertained after deadline
    - Don't send me assignments by using email, submit them on Google Class
    - Assignments **Viva** will also be conducted (during semester)
- **Attendance**
    - Don't mark Proxy of your friend (If he/she is your real friend ☺)
        - In case of proxy, **5 Absents** will be marked to that student
    - In case of emergency, get permission personally or send an email prior to the class
    - Attend all classes regularly, no attendance issue will be entertained in the end of semester
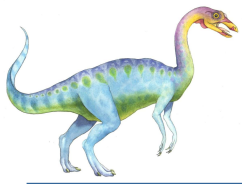
Are you ready !!!!

Lets Begin

Operating Systems

- How is RAM managed?

- How do you find something on disk?

- How do you know where to load it in RAM?

- How do you keep track of all running programs?

- Answer
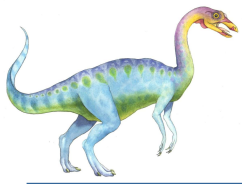 - This is what the operating system does

What is an Operating System?

- A program that acts as an intermediary between a user of a computer and the computer hardware

- Operating system goals:

  - Execute user programs and make solving user problems easier

  - Make the computer system convenient to use

  - Use the computer hardware in an efficient manner

Two Goals of Operating Systems

- Manage hardware resources
  - System operates smoothly, efficiently, reliably and securely
- Present abstract system model to programmer
  - Simple and convenient access to and control of resources

User Programs

present simple system model

Kernel

manage hardware resources

Hardware
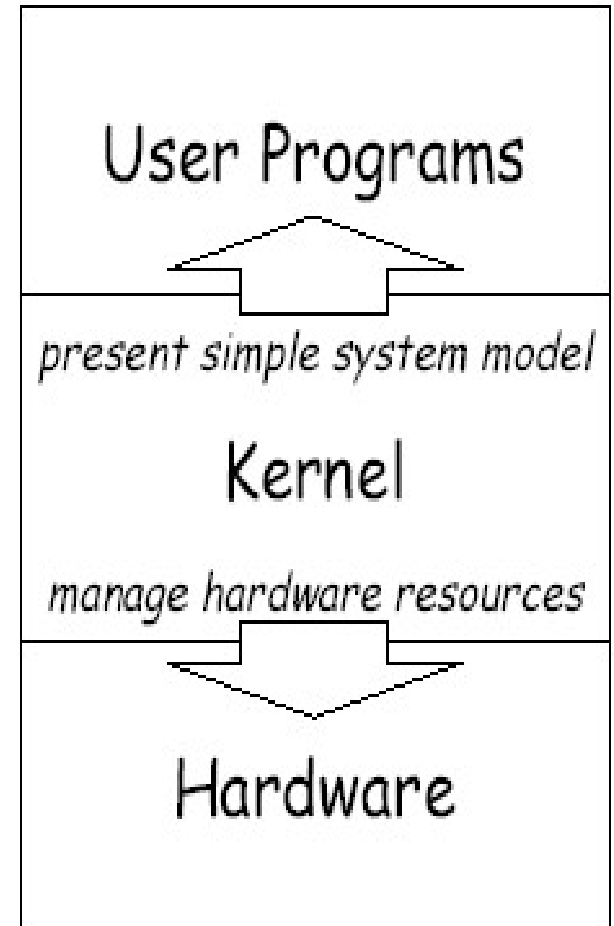
Manager/Coordinator of Resources

- Coordinates who gets what
  - "who": running programs
  - "what": resources
  - "when": scheduling time
  - "where": organizing space
  - "whether": limits, rights
- Goal: smooth system operation
 - efficiency, reliability, security
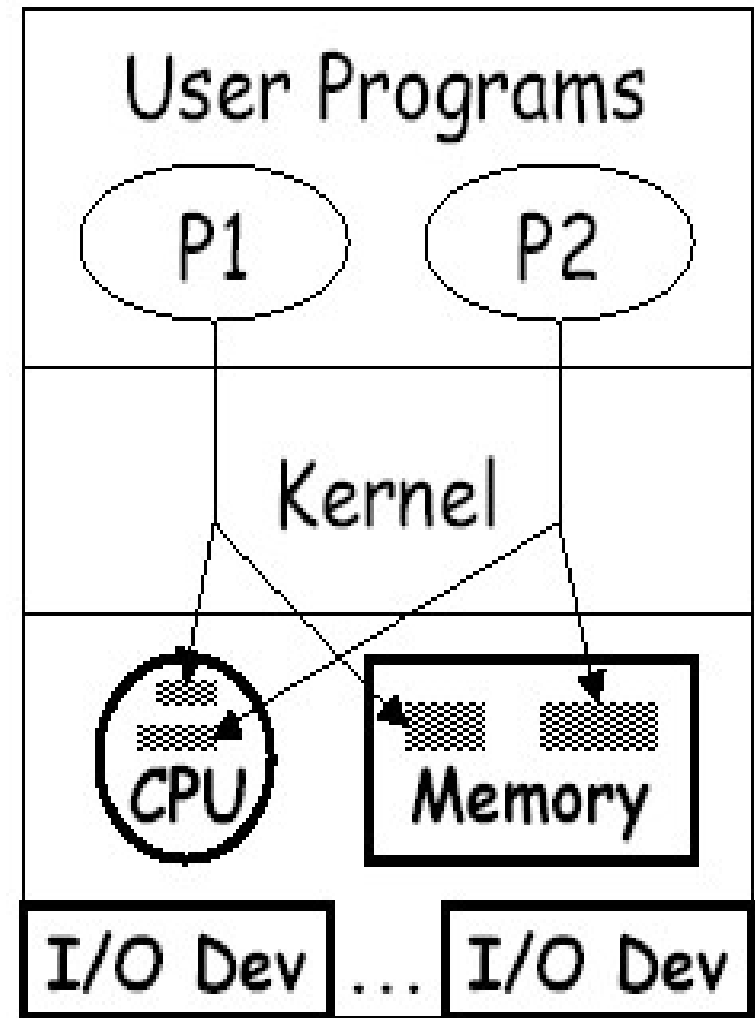
Resource multiplexing

- OS multiplexes resources in two ways:

- In time

  - Time multiplexing involves different programs taking turns in using the resource.

  - Example:
    - CPU scheduling
    - printer sharing

- In space

  - Space multiplexing involves different program getting part of the resource possibly at the same time

  - Example:
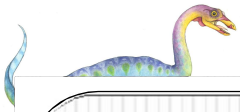   - memory is divided into several running programs

What If No Operating System?

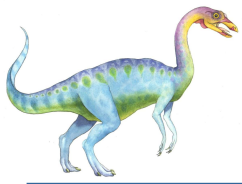| |
|---|
| User Program |
| Bare Hardware |
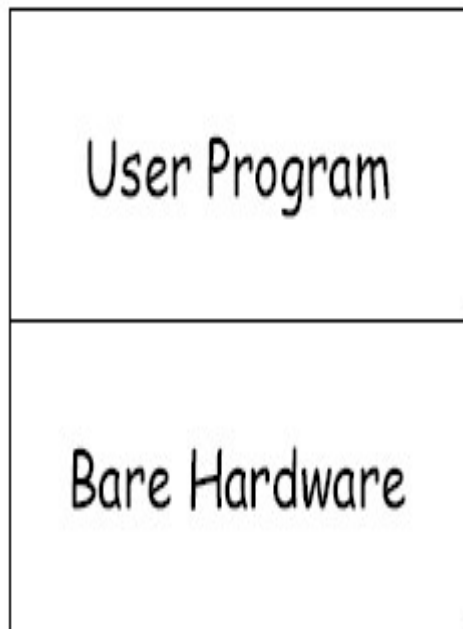
- All we have is bare hardware

- You want to run a program
  - How do you load it?
  - How do you run it?
  - What happens when it completes?

- Need at least some minimal OS to do these functions

…Lets go back to the "stone age" of computing…

Early Systems

- No Operating systems,
- your Data and your Programs
  - X = X + 1;
- As
  - Load   Register1, x
  - Load Direct   Register2, 1
  - Add   Register3, Register2, Register1
  - Store   Register3, x

- Time slots allocated to scientists
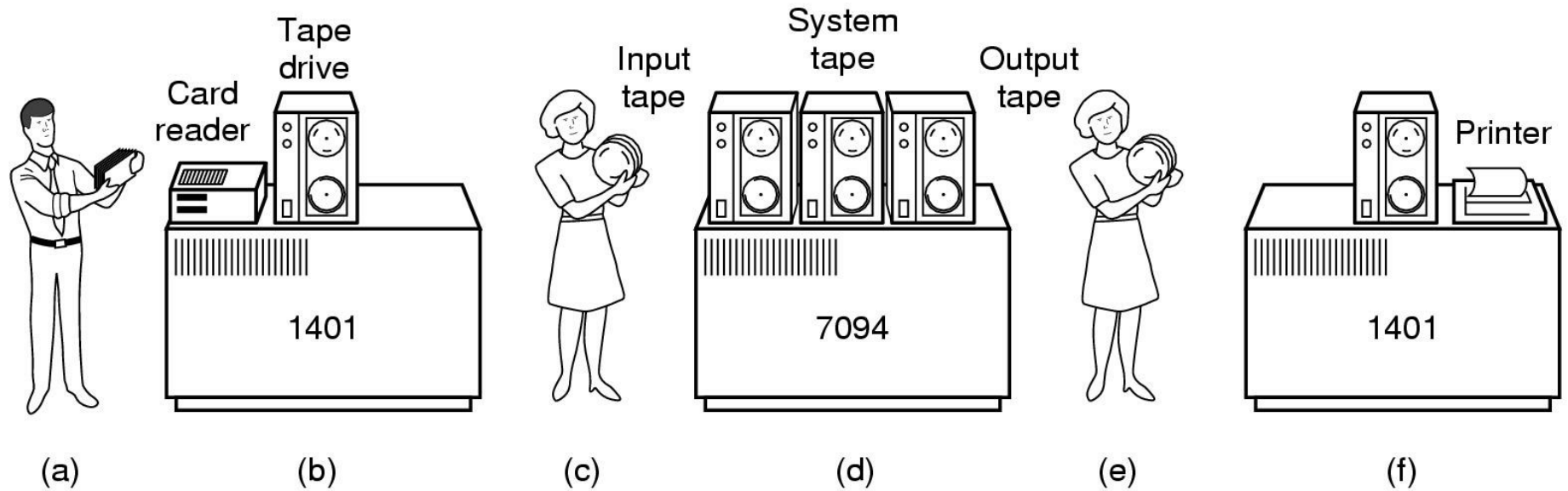- Enter programs using binary switches

Early Systems

- **Problem:**
  - computer remains idle while programmer sets things up
  - Poor utilization of huge investment

- **Solution:**
  - Hire a specialized person to do setup

- **Problem:**
  - Faster than programmer, but still a lot slower than the machine

- **Solution:**
  - Build a batch monitor

The Monitor

(a)   (b)   (c)   (d)   (e)   (f)

Early batch system
- bring cards to 1401
- read cards to tape
- put tape on 7094 which does computing
- put tape on 1401 which prints output

Simple Batch System

- The user submits a job (written on a card or tape) to a computer operator

- The computer operator place a batch of several jobs on a input device

- A special program, the monitor, manages the execution of each program in the batch

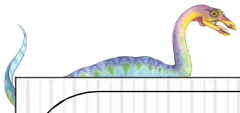- Resident monitor is in the main memory and available for execution

Why study operating systems?

- Point of describing change isn't "Look how stupid batch processing is" – it was right for tradeoffs of the time, but not anymore

- Point is: have to change with changing technology

- Situation today is much like it was in the late 60's

- OS's today are enormous, complex things
  - 100k's of lines (or >1M lines)
  - Windows NT is 20M lines
  - Windows2000 and Windows XP are about 40M lines.

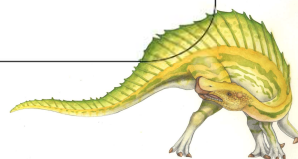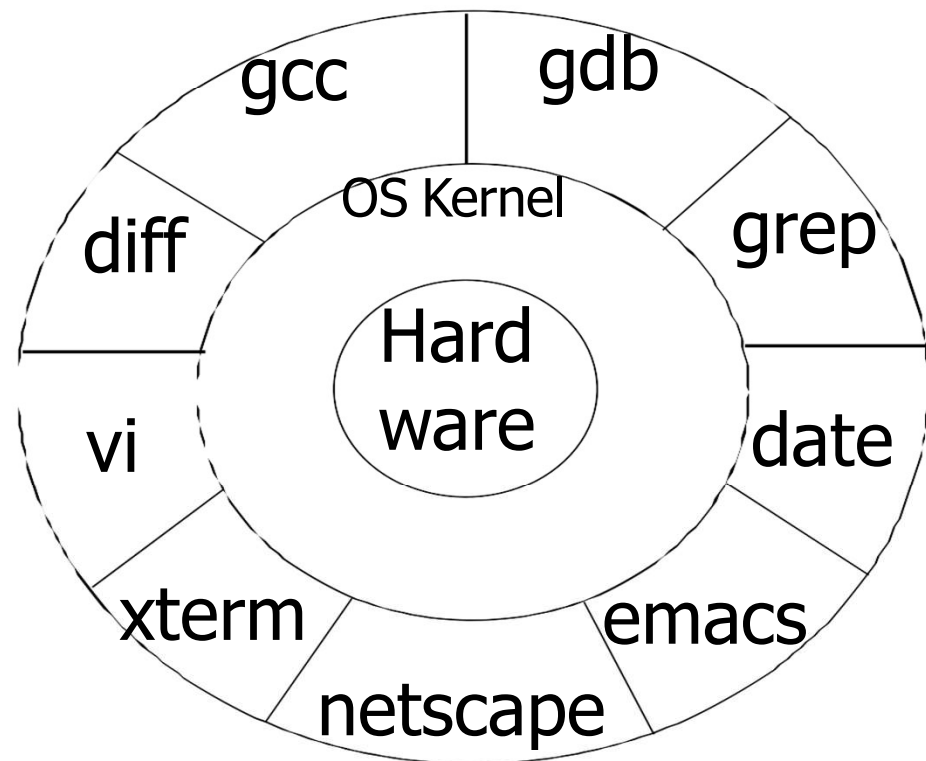- Key aspect of this course, understand OS's so we can simplify them!

OS
Overview

The Operating System controls the machine

User
Application
Operating System
Hardware

gcc

gdb

diff

OS Kernel

grep

vi

Hard
ware

date

xterm

emacs

netscape

A better picture

Application

System calls

Operating System

Machine instructions

Privileged instructions

Hardware

Many applications

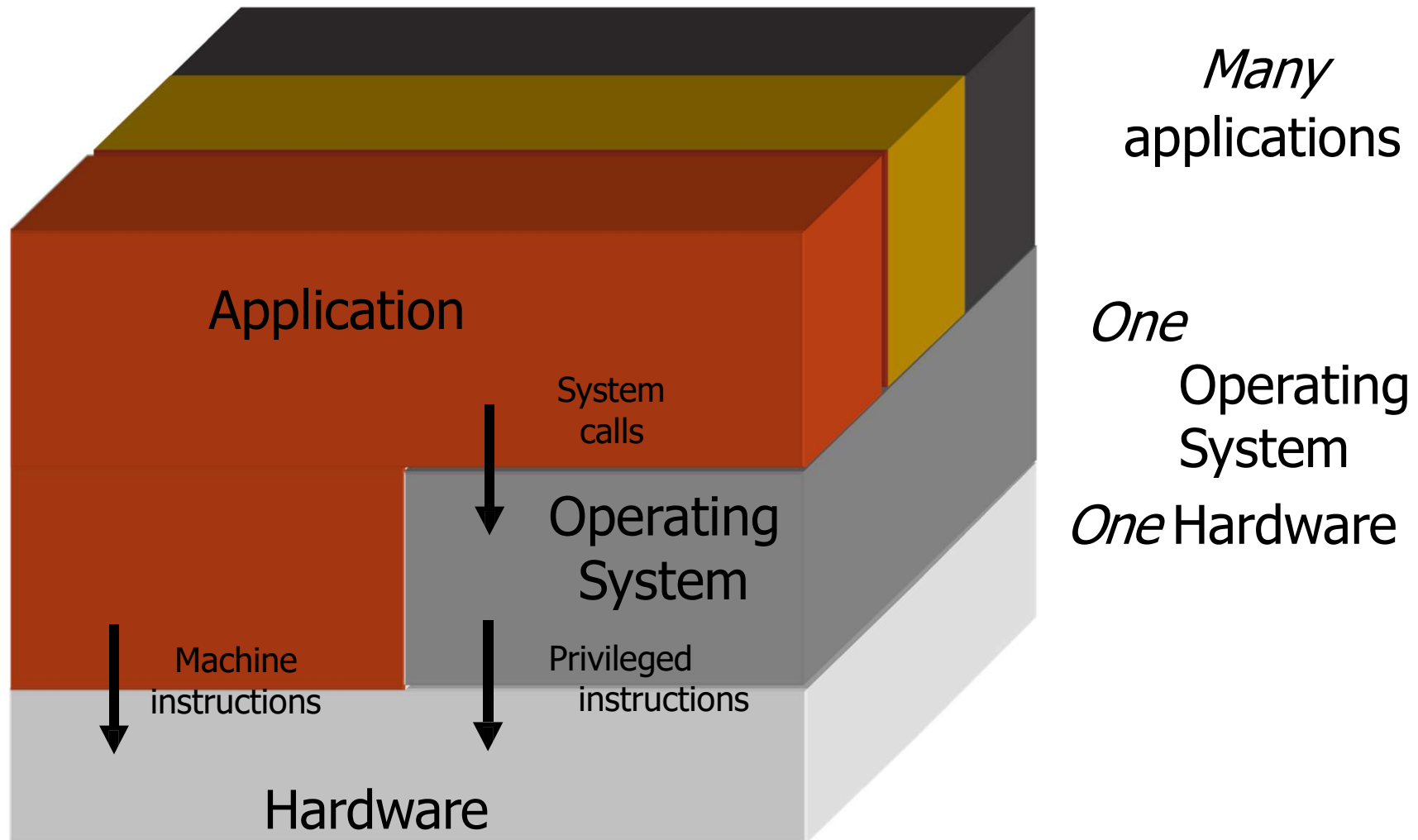*One* Operating System

One Hardware
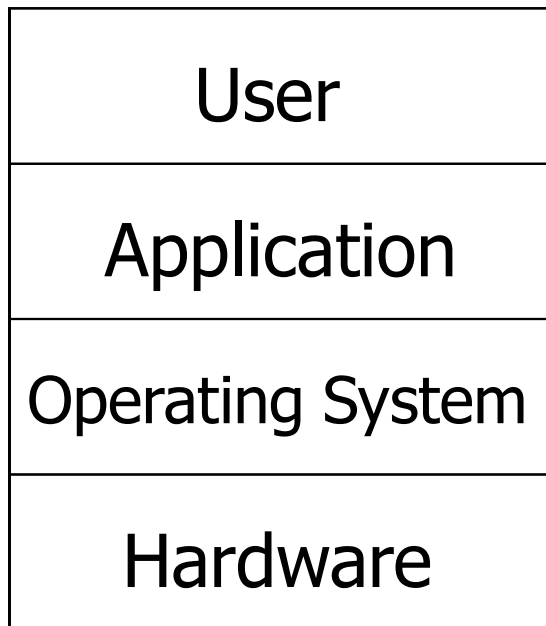
Operating System in Action

- OS is a program, just like any other program

- When you turn power on, *bootstrap* program is loaded from ROM

- Bootstrap program

  - Examine/check machine configuration

    - # CPUs

    - How much memory

    - # and type of HW devices

  - Build configuration structure describing the HW

  - Locates and Loads the OS

  - The control transfers to the OS
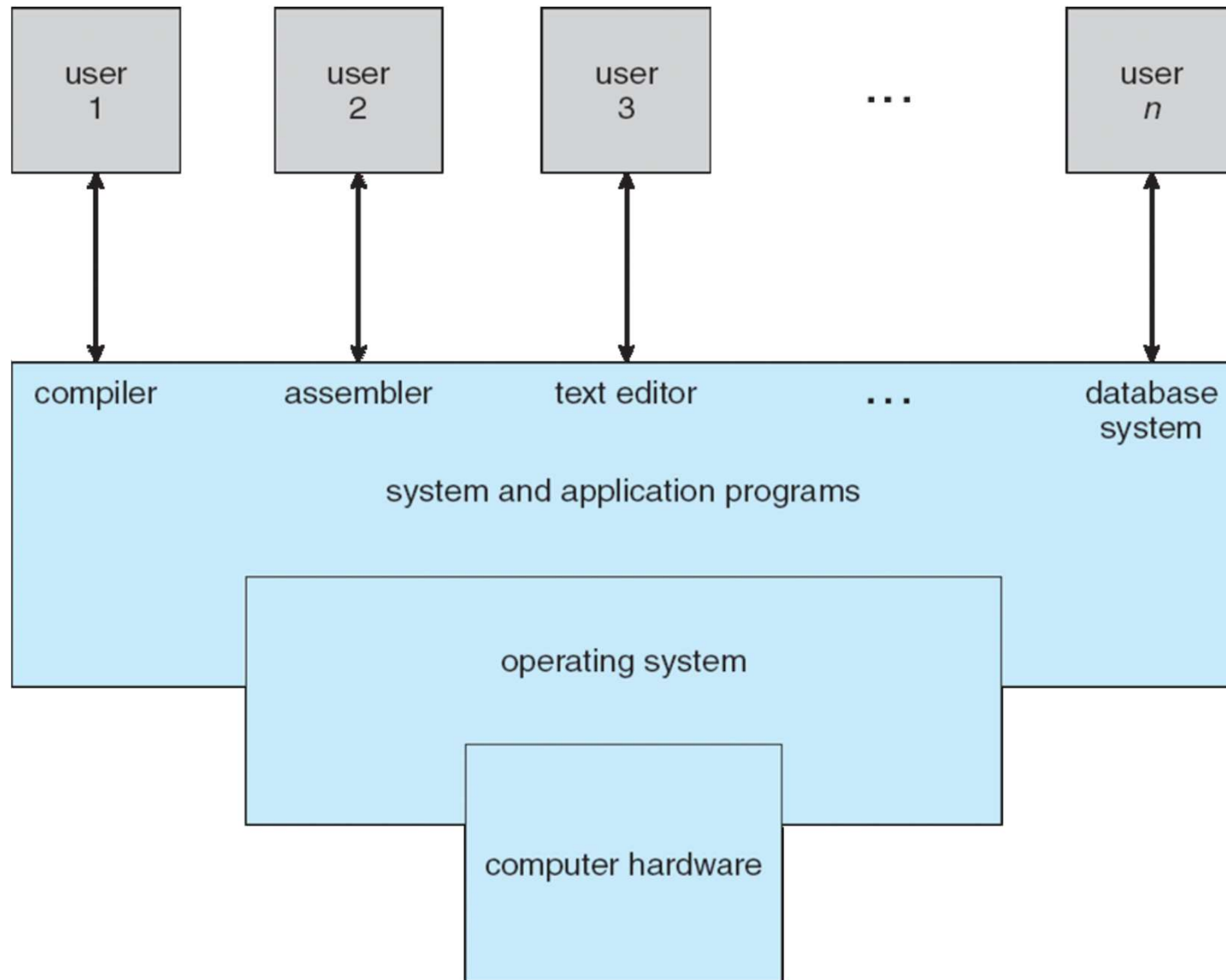
Operating System in Action

Operating System:

- Initialize kernel data structures

- Initialize state of HW devices

- Creates a number of processes to start operation
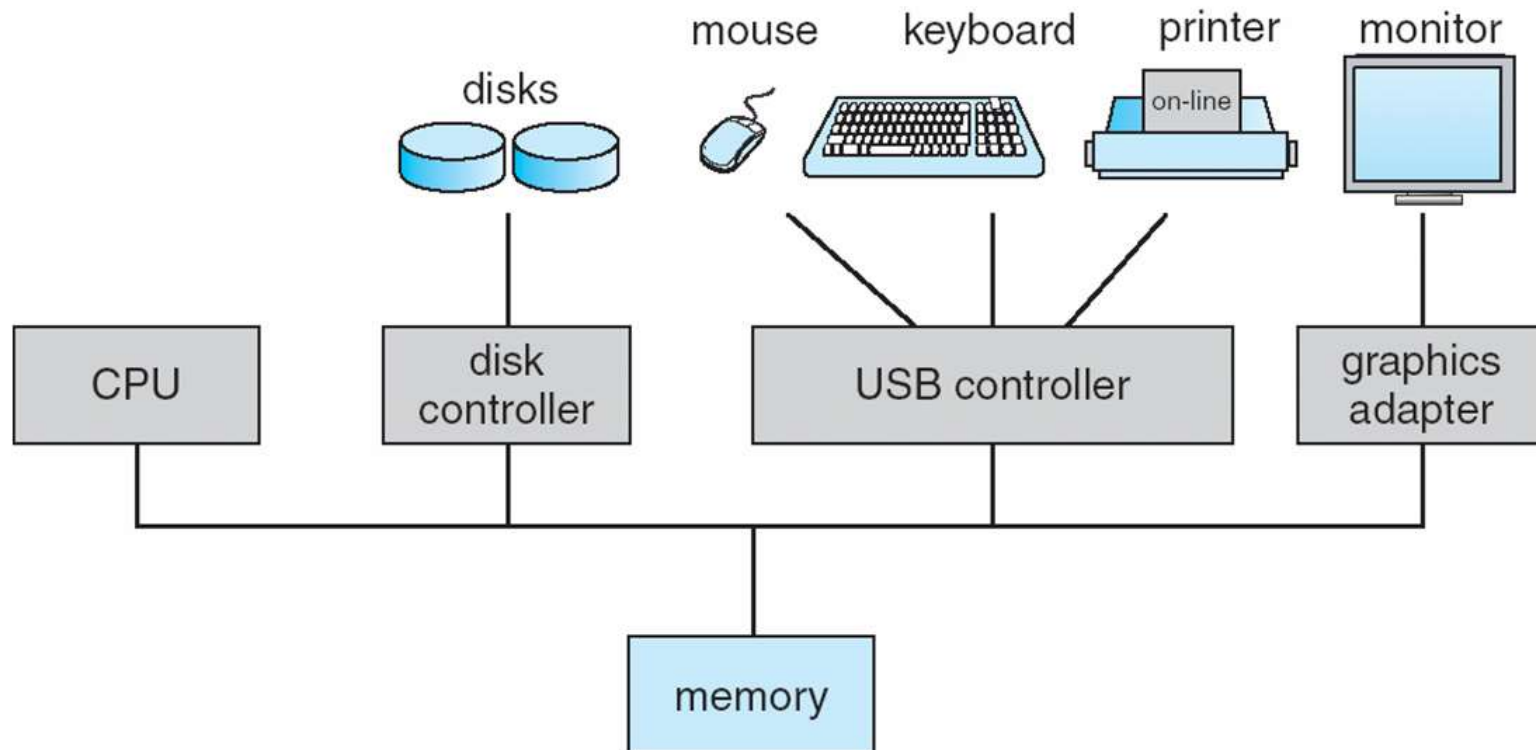
What is a Computer System?

Computer System Organization

- Computer-system operation
    - One or more CPUs, device controllers connect through common bus providing access to shared memory
    - Concurrent execution of CPUs and devices competing for memory cycles

Device Controllers

- I/O devices generally consists of two parts:
  - An Electronic component: Device Controller
  - A Mechanical component: The device itself
- Each device controller is in charge of a specific device
- It maintains
  - Some local storage buffer
  - A set of special purpose register
- Accepts command from the Operating System
- For example to read data from the device.

Device Controllers

- A Device controller might accept a command:
  - Read
  - Sector 11,206
- The device controller would:
  - Determine the current position of the head
  - Move the head to the required location
  - Accept data bit by bit
  - Store in a local buffer
  - Perform checksum on the data
- Controllers contain small embedded programs to carry out all this work.

Device Drivers

- Who would initiate the controller

- Who would determine read/write

- Who would tell the sector number

- A part of operating system called *Device Driver*

- Software or Hardware?

  - Software that talks to a controller

  - Gives it command

  - Accept Responses

- Same for all controllers?

- Different software for different type of controllers

- Each controller manufacturer has to supply a driver for each operating system it supports
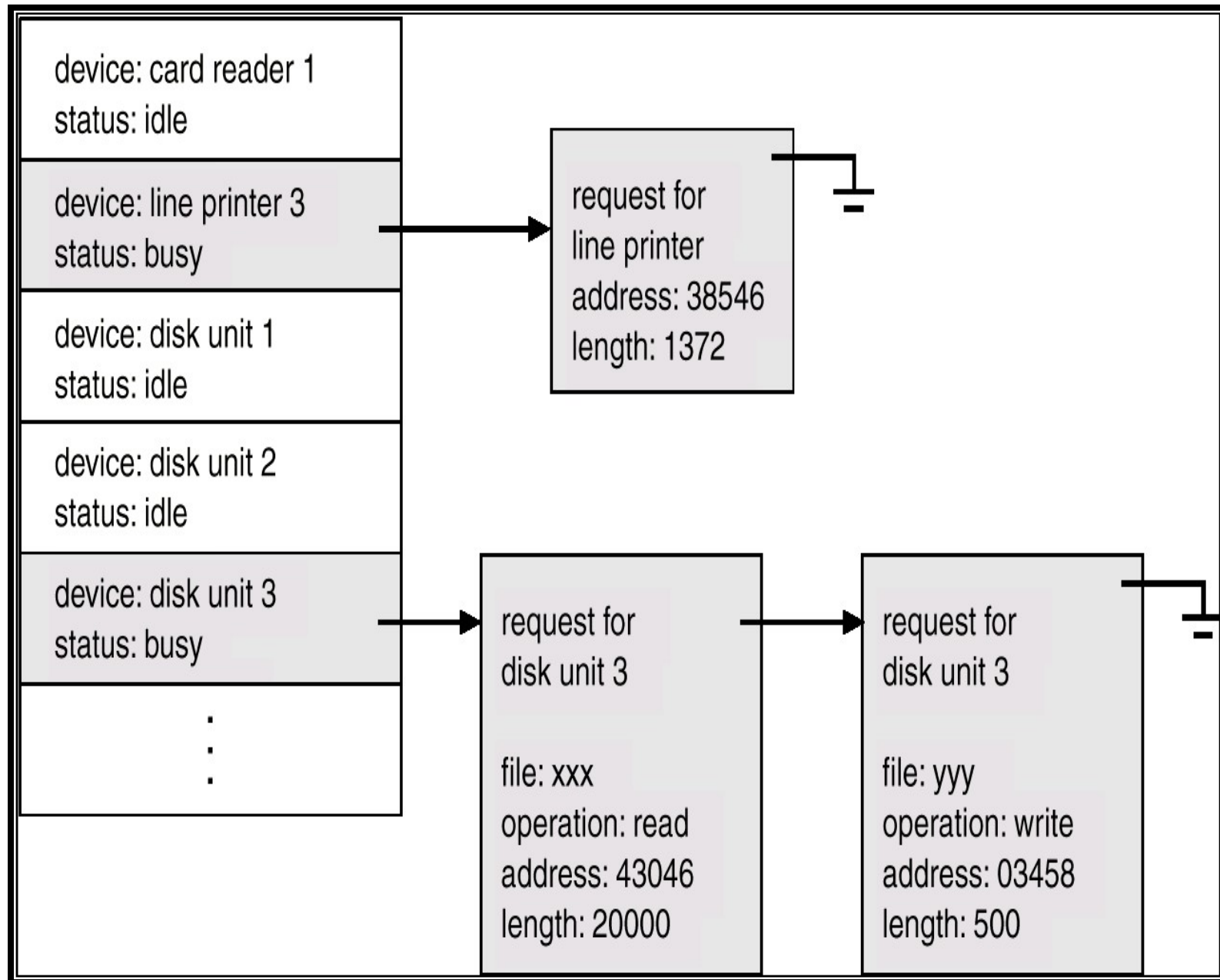
Device Status Table

☐ One entry per I/O device

☐ Keeps record of

   ☐ Device's type

   ☐ Address

   ☐ State

      ▸ Not functioning

      ▸ Idle

      ▸ Busy

☐ If the device is busy, the new request is saved in a queue

Device Status Table

References

- Operating System Concepts 8/9th Ed Chapter 1

- Modern Operating Systems 3rd Ed Chapter 1

- Operating Systems 6th Ed Chapter 2

  - By William Stalling

- http://en.wikipedia.org/wiki/Time-sharing

- http://en.wikipedia.org/wiki/Operating_system

- http://en.wikipedia.org/wiki/Distributed_computing

End