CS370 Operating Systems

Colorado State University
Yashwant K Malaiya
Spring 2020

Slides based on
- Text by Silberschatz, Galvin, Gagne
- Various sources
Operating Systems

• Part 1: How to do things
  – concurrently/in parallel

• Part 2: How to find stuff
  – Information in a many layered memory system

• Continued technological evolution
  – Techniques and challenges will evolve
  – Very high performance and capacity needed for modern applications: AI, Big Data
Technology Trends: Moore’s Law

Gordon Moore (co-founder of Intel) predicted in 1965 that the transistor density of semiconductor chips would double roughly every 18 months.

2X transistors/Chip Every 1.5 years
Called “Moore’s Law”

Microprocessors have become smaller, denser, and more powerful.

Moore’s law is dead? / not dead?
## Computer Performance Over Time

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniprocessor speed (MIPS)</td>
<td>1</td>
<td>200</td>
<td>2500</td>
<td>2.5K</td>
</tr>
<tr>
<td>CPUs per computer</td>
<td>1</td>
<td>1</td>
<td>10+</td>
<td>10+</td>
</tr>
<tr>
<td>Processor MIPS/$</td>
<td>$100K</td>
<td>$25</td>
<td>$0.20</td>
<td>500K</td>
</tr>
<tr>
<td>DRAM Capacity (MiB)/$</td>
<td>0.002</td>
<td>2</td>
<td>1K</td>
<td>500K</td>
</tr>
<tr>
<td>Disk Capacity (GiB)/$</td>
<td>0.003</td>
<td>7</td>
<td>25K</td>
<td>10M</td>
</tr>
<tr>
<td>Home Internet</td>
<td>300 bps</td>
<td>256 Kbps</td>
<td>20 Mbps</td>
<td>100K</td>
</tr>
<tr>
<td>Machine room network</td>
<td>10 Mbps (shared)</td>
<td>100 Mbps (switched)</td>
<td>10 Gbps (switched)</td>
<td>1000</td>
</tr>
<tr>
<td>Ratio of users to computers</td>
<td>100:1</td>
<td>1:1</td>
<td>1:several</td>
<td>100+</td>
</tr>
</tbody>
</table>

Anderson Dahlin 2014
Storage Capacity

- *Retail* hard disk capacity in GB
Course Resources

• Lectures: discussions, announcements, interaction
  – iClickers
• Canvas (Assignments, submission, grades)
• Piazza: discussions, announcements
• Webpage  http://www.cs.colostate.edu/~cs370
  – Home: Overview, contacts
  – Syllabus: Grading, Text, Responsibilities, Policies, Conduct
  – Schedule: Key dates, weekly schedules, slides, assignments, readings
ABOUT ME: Yashwant K. Malaiya

• My Research approach
  – Explore what has not been examined
  – Concepts contributed: Antirandom testing, Detectability Profile, New Vulnerability Discovery models, new Software reliability models

Areas in which I have published:

• Computer security
  – Vulnerability discovery
  – Risk evaluation
  – Assessing Impact of security breaches
  – Vulnerability markets

• Hardware and software
  – Testing & test effectiveness
  – Reliability and fault tolerance

• Results have been used by industry, researchers and educators
About me: Yashwant K. Malaiya

• Teaching
  – Computer Organization (CS270)
  – Operating systems (CS370)
  – Computer Architecture (CS470)
  – Fault tolerant computing (CS530)

• Professional
  – Organized International Conferences on Microarchitecture, VLSI Design, Testing, Software Reliability
  – Computer Science Accreditation: national & international
  – Professional lectures
  – Advised more than 65 graduate students ..
ABOUT ME: Jack Applin

• **Education**
  – BSE in Computer Engineering from the University of Michigan
  – MS in Computer Science from CSU

• **Experience**
  – Computer programmer at Hewlett-Packard for ~28 years

• **Service**
  – Faculty sponsor for MSTies Anonymous of Northern Colorado
About me: Jack Applin

• Teaching
  – Introduction to Unix (CS 155)
  – Introduction to C Programming I (CS 156)
  – Introduction to C Programming II (CS 157)
  – Foundations in Programming (CS 160)
  – Algorithms & Data Structures (CS 200)
  – Computer Organization (CS 270)
  – Software Development with C++ (CS 253)
  – Network and System Administration (CT 320)
Contacting us

• Instructors
  Yashwant Malaiya  Computer Science (CSB 356)
  Jack Applin  Computer Science (CSB 246)

• TAs, Office Hours in CSB 120
  Laksheen Mendis, Graduate TA
  Menuka Warushavithana, Graduate TA
  Sajeeb Roy Chowdhury, Undergraduate TA

• Preferred e-mail address cs370@cs.colostate.edu
  – The subject should start as CS370: ...
  – Specific email addresses: course web site

• Piazza:
  – Used for all updates/communications
  – Private posts will be seen by TAs/Instructors.
Topics we will cover in CS 370

• Processes
  – Processes and Threads
  – CPU Scheduling
  – Process Synchronization and Deadlocks

• Memory Management
  – Address translation
  – Virtual memory

• File System interface and management
  – Storage Management
  – File systems

• Virtualization
  – Data centers
  – Containers
Textbook

• Operating Systems Concepts, 10th edition
  Avi Silberschatz, Peter Galvin, and Greg Gagne
  etext package

• May also use materials from other sources including
  – Andrew S Tanenbaum, Modern Operating Systems
  – Thomas Anderson and Michael Dahlin, Operating Systems Principles & Practice
  – System Documentation, articles, news etc.
On the schedule page

• Topics that will be covered and the order in they will be covered
• Readings - chapters that I will cover
• May also see chapters mentions of other resources besides the textbook
• Schedule for when the assignments will be posted and when they are due
  — Subject to dynamic adjustment
Grading breakdown

• Assignments: 25%
  • Programming & written (note policies)

• Quizzes & interaction 20%
  – On-line and in-class (bring registered iClicker everytime)

• Mid Term: 20%

• Project: 10%

• Final exam: 25%

• You can only take the midterm/final for your section. The three sections are graded independently.
Grading Policy I

• Letter grades will be based on the following standard breakpoints:
  >= 90 is an A, >= 88 is an A-, 
  >=86 is a B+, >=80 is a B, >=78 is a B-, 
  >=76 is a C+, >=70 is a C, 
  >=60 is a D, and <60 is an F.

• I will not cut higher than this, but I may cut lower.

• There will be no make-up exams
  – Except for documented
    • required university event
    • acceptable family or medical emergency
Grading Policy II

- Plan: Every programming assignment will be posted 12-14 days before the due date. Written assignments will be posted 6-7 days before due date.
  - Every assignment will include specifications and will indicate it will be graded.
- Late submission penalty: 20% for the 24 hours and a ZERO thereafter.
- Detailed submission instructions posted on course website.
- Plan: Assignments will be graded within 2 weeks of submission
What will Quizzes and Tests include?

• I will only ask questions about what I teach, or ask you to study,
  – If I didn’t teach it, I won’t ask from that portion
  – Some on-line quiz questions about current state of technology may require you to search for an answer on the web

• If the concepts were covered in my lectures/slides/assignments
  – You should be able to answer the questions
  – You should be able to apply the concepts

• I will try to avoid questions about arcane aspects of some device controllers etc.
Exams & Assignments

• One mid-term
• The final exam is comprehensive, but more emphasis on the later part
• Quizzes: An on-line quiz almost every week. Iclicker interaction session time to time.
• Programming (5-6) / written (1-2) assignments
• Occasional help-sessions Wednesday 5:30 PM in CSB Including coming week
  – Attend or view recordings (required)
• Self exercises: Do them yourselves
Term Project

• Group based
  – Second half of the semester

• Options:
  – Research paper on current/developing technology
    • Paper, presentation, poster-session (dept)
    • Suggested topics will be announced
  – Development
    • IoT/Embedded system with sensor/communication
    • Design and evaluation needed
    • Demo presentations
Electronic devices in lecture room

• Use of Laptops and other electronic devices are not permitted.

• Exception: Permitted only in the last row, with the pledge that you will
  – not distract others, turn off wireless
  – use it only for class related note taking, which must be submitted periodically

• Laptop use lowers student grades, experiment shows, Screens also distract laptop-free classmates

• The Case for Banning Laptops in the Classroom

• Laptop multitasking hinders classroom learning for both users and nearby peers
Be kind to everyone

• You will be courteous to fellow students, instructor and the teaching assistants
  – Classroom, outside, discussion on Piazza

• Do not distract your peers
  – No chatting (except during iClicker sessions)
  – No eating
  – No cellphone use
Help me help you

• Surveys at the end of a class
• You will provide a list of
  – 2 concepts you followed clearly
  – 2 concepts you had problems keeping up
• Questions of interest for the majority of the class will be addressed in the next class
Help Sessions

• Some Wednesdays 5:30-6 PM, CSB 425 (to be verified)
• TAs will discuss key techniques and skills
  – Participation strongly encouraged
  – Slides and videos will be on the web site
  – You must be familiar with Help Session materials

• This week
  – C pointers, dynamic memory allocation
  – Needed for upcoming programming assignment
EXPECTATIONS

• You are expected to attend all classes.
• You must be present during the complete class
• Assignments have to be done by yourself individually. We will check.
• Expect to work at least 6-8 hours per week outside of class
  – Designing, coding and testing programs
  – Reviewing material from class
  – Do research for the project
• Concentrate in the class. The class have many new terms and concepts.
Expert view on How to fail this class?

• Believing that you can learn via osmosis

• Missing lectures
  – “If you don’t have the discipline to show up, you will most likely not have the discipline to catch up”
  – Procrastinating

• Get started on the assignments late. Note that they incorporate new concepts, including multiple processes and threads.
Interactions on Piazza

• You must sign up for Piazza
• You can have discussions with me, the GTA, and your peers
• But note
  – No code can be exchanged under any circumstances
  – No one takes over someone else’s keyboard
  – No code may be copied and pasted from anywhere, unless provided by us
• Appropriate use expected
From Operator to Operating System

Switchboard Operator

©UCB

Computer Operators
What is an Operating System?
What is an Operating System?

• **Referee**
  – Manage sharing of resources, Protection, Isolation
    • Resource allocation, isolation, communication

• **Illusionist**
  – Provide clean, easy to use abstractions of physical resources
    • Infinite memory, dedicated machine
    • Higher level objects: files, users, messages
    • Masking limitations, virtualization

• **Glue**
  – Common services
    • Storage, Window system, Networking
    • Sharing, Authorization
    • Look and feel
A Modern processor: SandyBridge

- **Package:** LGA 1155
  - 1155 pins
  - 95W design envelope
- **Cache:**
  - L1: 32K Inst, 32K Data (3 clock access)
  - L2: 256K (8 clock access)
  - Shared L3: 3MB – 20MB (not out yet)
- **Transistor count:**
  - 504 Million (2 cores, 3MB L3)
  - 2.27 Billion (8 cores, 20MB L3)
- Note that ring bus is on high metal layers – above the Shared L3 Cache
Functionality comes with great complexity!

SandyBridge I/O Configuration

- Proc
- Caches
- Memory
- Busses
- adapters
- Controllers
- I/O Devices: Disks, Displays, Keyboards
- Networks
- I/O Devices: Networks
- SandyBridge I/O Configuration
- SandyBridge I/O Configuration
- SandyBridge I/O Configuration
- SandyBridge I/O Configuration
- SandyBridge I/O Configuration
- SandyBridge I/O Configuration
Short History of Operating Systems

• One application at a time
  – Had complete control of hardware
• Batch systems
  – Keep CPU busy by having a queue of jobs
  – OS would load next job while current one runs
• Multiple programs on computer at same time
  – Multiprogramming: run multiple programs at seemingly at the “same time”
  – Multiple programs by multiple or single user
• Multiple processors in the same computer
• Multiple OSs on the same computer
Early processors (LC-3 is an example)

- Instructions and data fetched from Main Memory using a program counter (PC)
- Traps and Subroutines
  - Obtaining address to branch to, and coming back
  - Using Stack Frames for holding
    - Prior PC, FP
    - Arguments and local variables
- Dynamic memory allocation and heap
- Global data
One Processor One program View

- External devices: disk, network, screen, keyboard etc.
- Device interface: Status and data registers
- **User and Supervisor modes** for processor
- I/O
  - Device drivers can use polling or **interrupt**
  - Interrupts need **context switch**
  - I/O done in supervisor mode
  - **System calls** invoke device drivers
What a simple view doesn’t include

• Cache between CPU and main memory
  – Makes the main memory appear much faster
• Direct memory access (DMA) between Main Memory and Disk (or network etc)
  – Transfer by blocks at a time
• Neglecting the fact that memory access slower than register access
• Letting program run \textit{concurrently} (Multiprogramming) or with many threads
• Multiple processors in the system (like in Multicore)
• Multiple OSs in the same system
Information transfer in a system

- **CPU Registers – (Caches) - Memory**
  - CPU addresses memory locations
  - Bytes/words at a time
  - We will see some details

- **Memory – (Controllers hw/sw) - external devices**
  - Chunks of data
  - External devices have their own timing
    - DMA with interrupts
  - Disk is external!
System I/O (Chap 13)

Diagram illustrating the system architecture with components such as monitor, processor, central brain, graphics controller, bridge/memory controller, cache, memory, SCSI controller, PCI bus, IDE disk controller, expansion bus interface, keyboard, parallel port, and serial port.
I/O Hardware (Cont.)

• I/O Devices usually have registers where device driver places commands, addresses, and data
  – Data-in register, data-out register, status register, control register
  – Typically 1-4 bytes, or FIFO buffer

• Devices have addresses, used by
  – Direct I/O instructions
  – Memory-mapped I/O
    • Device data and command registers mapped to processor address space
I/O Transfer rates MB/sec

- system bus
- HyperTransport (32-pair)
- PCI Express 2.0 (×32)
- Infiniband (QDR 12X)
- Serial ATA (SATA-300)
- Gigabit Ethernet
- SCSI bus
- FireWire
- hard disk
- modem
- mouse
- keyboard
Acknowledgments

• Jack Applin for his editing comments on the slides in S2020 for enhancing readability.

• Past CS370 instructors, specifically Shrideep Pallickara for contributions to the class including materials and methods