Announcements

Final Project: Deadlines

• Project idea known by now
• Wed (12/7): Project draft to Learn@UW dropbox
  – Whatever you have completed
• Due December 12 – In-class Demos
  – Demo with TA for grading – last week

Upcoming Office/Lab hours

• Today 11:00 – 12:00 (Instructor 7375)
• Tuesday in 1370: 12:30 – 2:25 Thea
• Tuesday 7375: 2:30 – 4:30 Instructor

Extra Credit: Fill out survey for College Board

• Screenshot to Dropbox HW11-Extra Credit (1/2 hw grade)

Users Run Many Applications Simultaneously

Expect all to be running, doing work for you...

Today’s Mystery

How does computer run multiple applications on one set of hardware?

• CPU: processing unit (ALU + set of registers) and control unit (program counter)
• Main Memory (RAM)

How are applications going to share?
**What is an Operating System?**

Operating System (OS):
- Software that converts hardware into a useful form for many different applications.
- Very complex: millions of lines of code, 1000 person-years.

**Roommate Scenario**

Imagine two roommates sharing a double dorm room (1 bath).

What properties must hold to give illusion of own room?
- Other person doesn’t vandalize their stuff.
- Other person doesn’t look through their personal stuff.
- Can use bathroom whenever you need.

Some properties about:
- Protecting your stuff
- Getting to do what you want

**How does OS help applications share hardware?**

OS gives each application illusion that it is only one running on hardware.
- Manage hardware resources for applications.

**How does this match Computer?**

Multiple applications sharing same hardware.

**General Requirements**
- Protecting your stuff = Protecting data that resides in Memory.
- Getting to do what you want = Running on CPU when you want.

**Specifics**
- No vandalism of stuff = Another app can’t overwrite your data.
- Can’t look through stuff = Another app can’t read your data.
- Can use bathroom when needed = Run on CPU when app has work to do (not when sleeping).

Must handle misbehaving apps – Before harm occurs:
- What if roommate won’t leave bathroom????
- Must have way to remove them against their will!
Terminology

OS runs/executes "processes" not applications
- An application may be composed of multiple processes

What is a process?
Execution stream (i.e., instructions) in context of process state (i.e., data)
- What you want to do, plus your stuff

Find processing by running "ps" (Unix-based)
- More processes running than you might expect!

Hardware Resources

1) How to share memory?
2) How to share CPU?

Terminology

Multi-programming: Multiple processes resident in memory at same time (run "concurrently")
- Same as multi-tasking

Opposite: Uni-programming
- Only one process resident at a time
- Examples: First systems and DOS for PCs

Not multiprocessing: Multiple processors

Advantages: Better user convenience and performance

How to Share Memory?

Illusion provided by Operating System?
- Each process has all of physical memory to itself

Reality: Reside in physical memory at same time
Technique: Space sharing

Logical View of Address Space
- Everything process can address thru memory including data and code
Challenges with Sharing Memory

1) Ensure one process cannot r/w another process’s memory
   - OS and HW cooperate to implement protection
     - Translate memory references from logical to physical addresses

2) Not enough physical memory for all address spaces
   - What can OS do when not enough memory for all processes?

What do you do when too much important stuff?

Keep your extra stuff somewhere else...
   - Keep things back home at parents’ house

Computer system: Move data to disk if can’t fit
   - Thrashing or paging: Spending time moving data you care about between main memory and disk
   - Move stuff rarely use

How to Share CPU?

Process alternates b/wn CPU and I/O
   - I/O: wait for user input
   - Analogy: Alternate b/wn bathroom and bedroom
   - Time sharing: Switch quickly between processes

OS and HW together perform context switch
   - Change contents of registers and Program Counter (PC)
     - Stored in memory when not running (only OS can read this memory)
   - Change active address space in memory
     - One process should not be able to read data of another process!

Perform context switches at different points
   - When one job waits for I/O, switch to new job
   - When one job has been using CPU too long, switch
     - Prevents one process from hogging CPU
How to Share CPU?

If only one process wants CPU, no problem

What do you do if multiple processes want CPU at the same time?

OS must also implement policy
  • Many processes want to run, but which should run when??

Easiest Policy?

First-come-first-served (FCFS)

How to implement this policy?
  • Customers take ticket when arrive, serve next number
  • Customers add to end of line, serve next customer in line

Why do you think this a good policy?
  • Easy to implement
  • Intuitively Fair: Earlier you arrive, sooner you get service

Why could it be bad?
  • Treats all customers identically but may have different requirements

Example: Service at a Deli

Many customers waiting for service at deli...
In what order should customers be handled?

Different Requirements?

1) Some costumers have a deadline

2) Some customers are more important than others

3) Some costumers have short orders, others very long orders
1) Scheduler for Handling... Deadlines?

**Earliest Deadline First**
- Ask everyone when need to be done by
- Serve costumer with next deadline (search for min!)

**Examples in Real World?**
- Sometimes in long lines for airline check-in

**What is good?**
- Everyone finishes by when they need to

**What is bad about this approach?**
- Not fair: Works best when everyone works together
- Needs knowledge and trust: When is your real deadline?
- Impossibility: Might not be able to meet all deadlines

---

2) Scheduler for Handling... Important Customers?

**Priority-based scheduling**
- Allow important customers to move to front of line

**Examples in real life?**
- First-class in airlines, Fast Pass at Amusement Parks

**Advantages?**
- Give fastest service to most important customers (make them happiest)

**Disadvantages?**
- Less important customers can starve
  - Might never receive service if many important customers keep arriving
  - Extreme of "unfair"
- Determining who is "important" can be difficult
  - Spend the most money? Influence the most other people? Angriest?

---

**Earliest Deadline First**

**In Computer Systems?**
- Used for "real time" and "embedded" applications
- Control system must periodically perform different tasks
  - Check different sensors (temp, speed, location, battery life)
  - Adjust different controls (rotation, power)

**How to determine deadlines?**
- Every task needs to run periodically at same interval
- Next deadline = last time ran + interval
- Some sensors and controls more important than others
  - Check/control them more frequently...

---

**Priority Scheduling**

**In Computer Systems...**
Which processes should be given high priority?

- Give higher priority to system processes
  - Responsible for keeping machine running

- Give higher priority to "interactive" processes
  - Processes user is currently "interacting" with
    - Give priority to which process they are typing to
    - Give priority to which process is creating output
3) Scheduler for Handling... Short Jobs?

“Shortest Job First” (SJF)
- Figure out which customer has shortest order
- Put shortest orders go to front of line (search)

Examples in Real World?
- Decide to interrupt counter person with question...
- Separate lines for “10 items or less”

Advantages
- Creates optimal schedule for average waiting time
  - Minimizes average waiting time over all customers
  - Moving short job before long job:
    - Improvement in wait time of short job > Penalty to long job

Disadvantages
- Customers with many items can starve; unfair
- How can you tell how long job will take? Incentive to lie!
  - Must solve for this to work in computer systems!
  - How???

Big Idea:
Use Past to Predict Future

Processes behave in future similarly to past (just like people?)
- Did this process use CPU for small time in the past?
- Use info to schedule short CPU bursts

Remember: Process alternates btwn CPU and I/O (e.g., wait for user input)

Today’s Summary

Operating System: Software that manages hardware
- Provides illusion to each process that it’s only one running
  - Context switches CPU across processes (Time share)
  - Protects memory across processes (Space share)
- Scheduling policies for CPU:
  - First-come-first-served (FCFS), Earliest-deadline-first,
    Priority-based, Shortest-Job-First (SJF)

Upcoming Office/Lab hours
- Today 11:00 – 12:00 (Instructor 7375)
- Tuesday in 1370: 12:30 – 2:25 Thea
- Tuesday 7375: 2:30 – 4:30 Instructor

Extra Credit: Fill out survey for College Board
- Screenshot to Dropbox HW11-Extra Credit (1/2 hw grade)