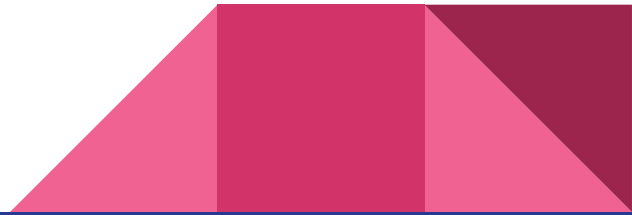
Parallel/Distributed, Internet, and Fault Tolerance

Your 15-110 TAs!

CPU and Running Programs

CPU (Central Processing Unit) is the part of a computer's hardware that runs actions taken by the program

- Interaction with memory
- Scheduler to run multiple programs
  - Decides which program gets to use the CPU
- **Multitasking** (fake concurrency)
  - Alternating between steps rapidly
  - Appears to run at the same time
  - Steps chosen to maximize **throughput** to the user



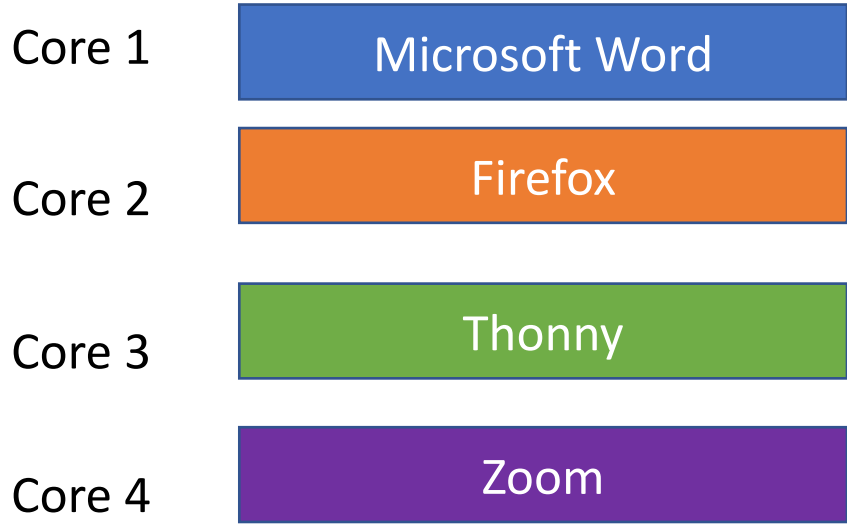
Multiprocessing and Parallel Programming

Multiprocessing – multiple CPUs

Multiprocessing is running multiple actions at the same time on a single computer using multiple CPUs.

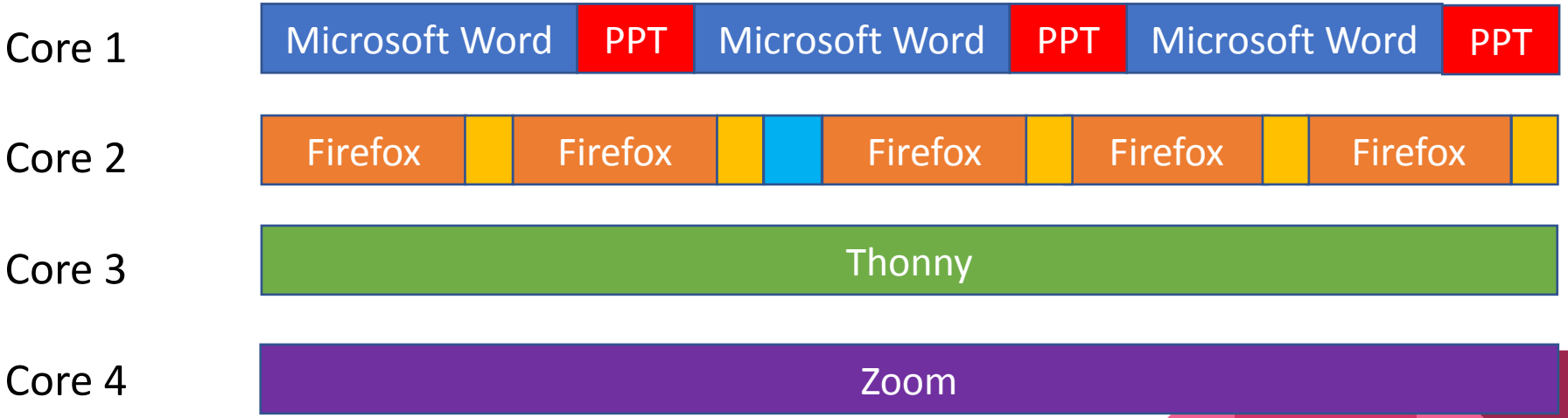
Scheduling with multiprocessing

- Run application simultaneously on different CPUs
  - Each work independently



Multitasking with multiple CPUs

Computers use multitasking and multiple CPUs to run more applications that appear to run concurrently



Locking and Yielding

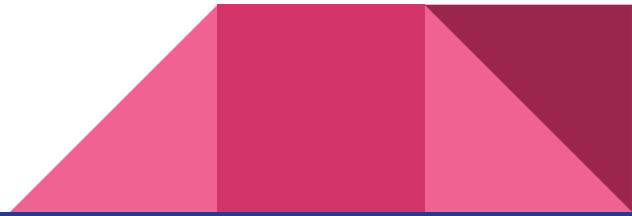
How do we avoid mix-ups when two programs want to use the same resource?

Programs put a **lock** on a shared resource while they're using it.

Yields when done with resource.

Deadlock

- Happens when programs lock the resource they're using and wait for the next resource to become available
- Will wait forever!
- Resolve this by communication



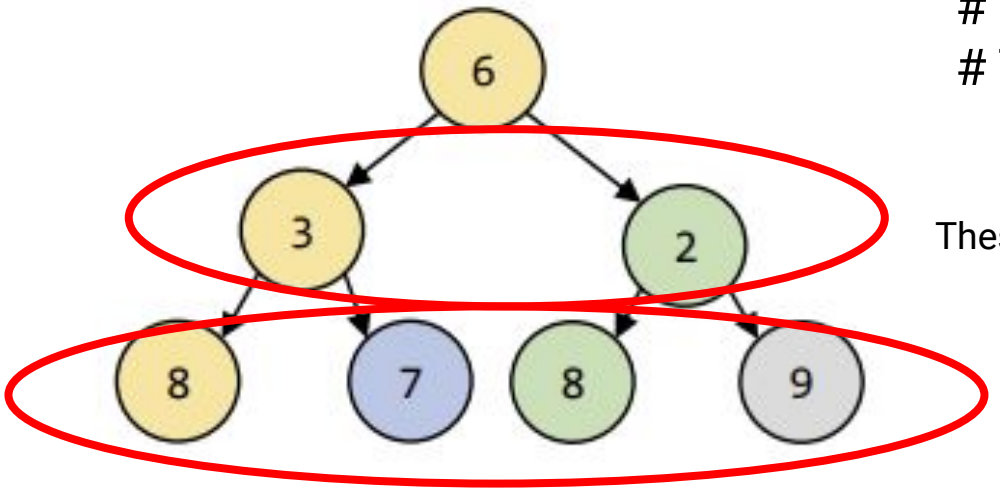
Parallel Programming

Parallel programming – runs single process on multiple cores

Determine efficiency of a parallel algorithm by comparing the total number of steps to the number of time steps

- **Total number of steps** = number of actions taking place
- **Time steps** = number of steps taken over time (steps can be merged into one when they happen at the same time)

Total Steps vs. Time Steps: Tree Example



Time Steps = 2
Total Steps = 7

These tasks happen at same time.

Pipelining

Pipelining simplifies parallel processing

Pipelining involves having one “worker” repeat the same task, similar to an assembly line.

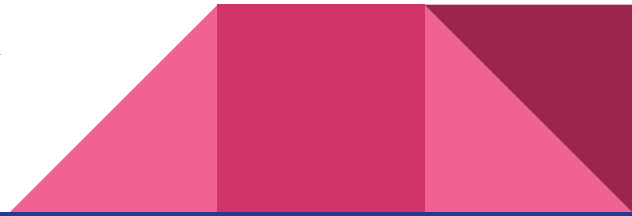
Example of making a sandwich:

1. Spread peanut butter on one slice of bread
2. Spread jelly on the other slice of bread
3. Combine and cut sandwich

To make **4 sandwiches, there are 12 steps and 6 time steps.**

Pipelining rules:

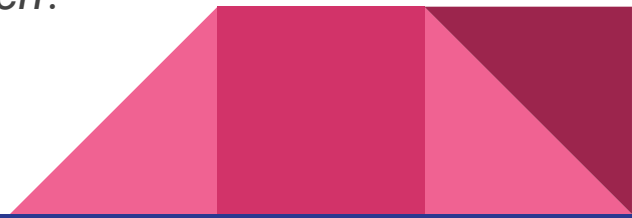
- One task must be completed to move on to the next task
- Length of time depends on the longest step



Pipelining

- Like an assembly line. Tasks are split across different cores (workers). When given a piece of data, it executes the step, then passes result to next core.

- **Pipelining Example**: You are organizing a science fair with 3 judges. One judge examines experiment subject (S), another judges experimental design (D), and another judges result visualization (V). Each judge evaluates a project for two minutes, and one poster project must be evaluated in all 3 categories. *How many posters can be evaluated with this approach?*



0 min	2 min	4 min	6 min	8 min	10 min	12 min	14 min	16 min	18 min
S	S	S	S	S	S	S	S	S	S
	D	D	D	D	D	D	D	D	D
		V	V	V	V	V	V	V	V

So, 8 posters can be judged using pipelining.

Distributed Computing

Distributed Computing

Distributed computing uses multiple machines, while multiprocessing uses multiple CPUs.

- Computer assigned smaller task
- CPUs inside computers assigned even smaller task

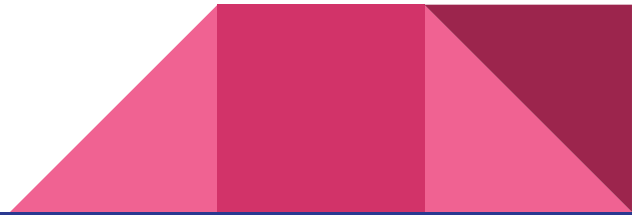
MapReduce

MapReduce breaks up the data into smaller sections, rather than breaking down the steps.

Mapper: takes piece of data, processes it, finds partial result

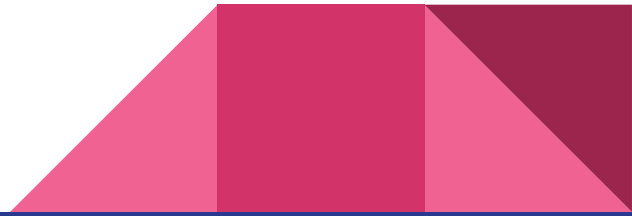
Reducer: takes a set of results and combines them

Manager: moves the data through the processes (splits data to send to mappers, gets results and sends the combination to the reducer, outputs the final result from the reducer)



MapReduce Person

- **MapReduce Example**: You are organizing a science fair with 3 judges. One judge examines experiment subject (S), another judges experimental design (D), and another judges result visualization (V). Each judge evaluates a project for two minutes, and one poster project must be evaluated in all 3 categories. *How many posters can be evaluated with this approach?*



0 min	2 min	4 min	6 min	8 min	10 min	12 min	14 min	16 min	18 min
S	D	V	S	D	V	S	D	V	S
S	D	V	S	D	V	S	D	V	S
S	D	V	S	D	V	S	D	V	S

So, 9 posters can be judged with this approach.

How the Internet Works

The Internet

The internet is a decentralized **network of computers** (a graph of computers!)

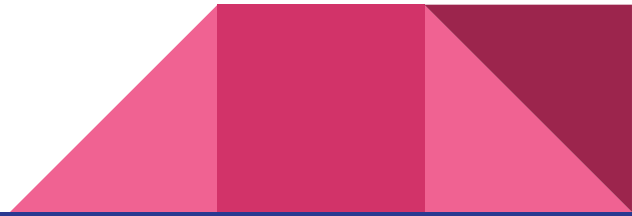
- **Routers** are the core of the internet that help send data around the world
- **ISPs** manage local connections to connect a computer to the internet
- Ex: Computers – ISP — Router – Router – ISP – Computers

Website Representation

Websites use **HTML** (HyperText Markup Language) to display visual content.

URLs are nicknames, IP addresses are real names of websites

- IP addresses uniquely identify websites and computers
- IP addresses for specific websites are static, IP addresses for computers that go online and offline frequently are dynamic



Retrieving a Web Page

- Computer obtains IP address by obtaining it from the DNS Server, which contains a mapping of all URLs to IP Addresses.
- Once your computer knows IP address, sends a request for the page to the IP Address (HTTP)
- The request is like a message that is sent as a “packet”
- When website receives the request, it usually needs to send back a webpage
  - Webpages are split into multiple packets and are sent back through routers to your computer
  - Can take different paths and arrive in different orders
- Your web browser assembles all the received packets into webpage and displays it to you

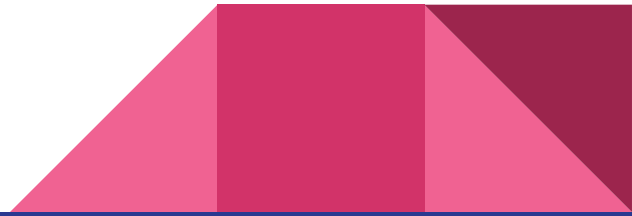
Fault Tolerance

Packet Fault Tolerance

- Missing packet: Browser sends request for new packet
- Corrupted packet: Packet has checksum for computer to check it’s not corrupted. If corrupted, requests new packet.

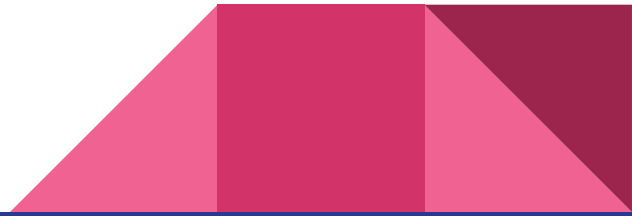
Network Fault Tolerance

- Computer goes down: ISP sees computer go offline, holds received data until you’re back
- Company’s website (server) goes down: traffic to that server gets re-routed to different server. If all of company’s servers are down, website goes down.



Network Fault Tolerance (cont’d.)

- Router goes down: traffic is sent to other routers instead
- DNS Server goes down: request gets sent to different DNS Server
- ISP goes down: Lose connection to entire internet.



Cyber Attacks

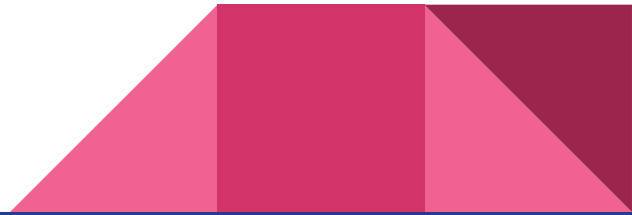
Attacks

Man in the Middle Attacks

Adversary sets up a router that pretends to be a normal router in the network. Evil router intercepts packets being sent to different destinations. Only possible if packets are not encrypted.

Distributed Denial of Service (DDOS) Attacks

Adversary sends or receives huge amount of data to/from server within short time frame. Overwhelms server and makes it impossible to respond to actual user requests, so it appears down for normal user.



Other things to review:

- HW Problems
- Lecture Notes
- Practice Tests
- Small Group Problems
- Recitation Problems
- These slides