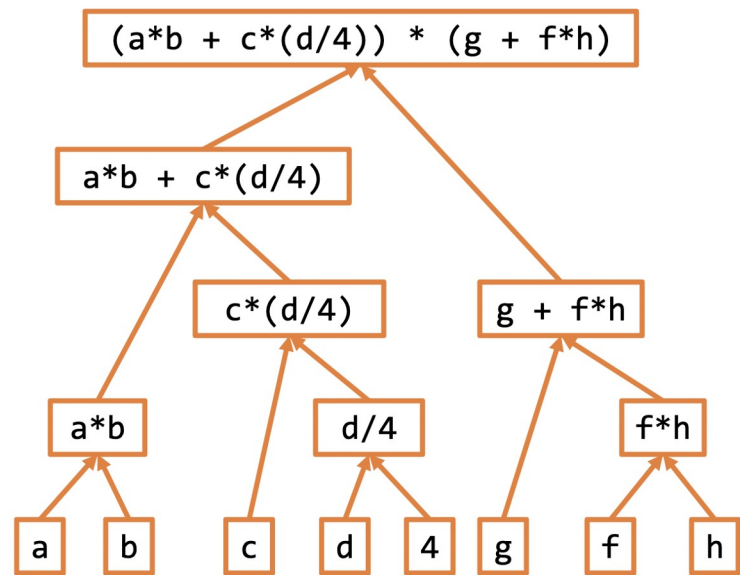# Quiz 5 Review Session

There are four types of concurrency.

1. Circuit-Level Concurrency: concurrent actions on a single CPU
2. Multitasking: seemingly-concurrent programs on a single CPU
3. Multiprocessing: concurrent programs across multiple CPUs
4. Distributed Computing: concurrent programs across multiple computers

Concurrency Tree

- Tree that shows how operations can be broken down into time steps
  - Nodes on the same level are simultaneous actions
  - Total number of steps: number of non-leaf nodes
  - Time Steps: number of non-leaf levels

```
(a*b + c*(d/4)) * (g + f*h)

a*b + c*(d/4)

c*(d/4)          g + f*h

a*b        d/4              f*h

a   b    c    d    4    g    f    h
```

Problems with Multiprocessing

- Deadlock: occurs when two or more processes are all waiting for a resource that another group holds → caused by locking and yielding resources
- Difficulty of design: must design algorithms that work across multiple processes and computers

Pipelining

- increases efficiency of repeated operations by using sub-steps simultaneously
- like an assembly line: the cores start new computations while others are still in progress
- each core has one task
- the order in steps is important
- the length of time depends on the longest step
- most useful when shared resources are limited

MapReduce Pattern

- algorithm for organizing parallelized programs
- takes large data set and breaks up data across cores
- **mapper**: takes piece of data and finds partial result
- **reducer**: takes set of partial results and combines them
- **manager**: moves data through mapper and reducer, outputs final result

The Internet

- **decentralized** global network of connected computer networks
- **routers**: take in data and send to certain location
    - connected by cables or wi-fi
- **ISPs**: connect user's computer to Internet
- **browser**: application that receives data from Internet and organizes into webpage
    - receives text and turns into visual content using **HTML protocol**
- **URL** is like a nickname for website, and **IP address** is like the real name
    - ex. URL: google.com, IP address: 172.217.9.206
    - IP address can be **static** - for specific websites, or **dynamic** - assigned to different computers at different times

The Internet

- **DNS server**: computer that maintains a mapping of all URLs to IP addresses
  - finds requested IP address and sends it back to ISP through routers
- **protocol**: standard conditions that need to be met for request to be fulfilled
  - HTTP is the standard protocol for requesting information from a website
- **packet**: stores the data and records destination and return addresses (IP addresses)
  - takes different paths to get to and from IP addresses
- **the cloud**: computers that are connected to the Internet
  - exceptionally fault tolerant

The internet is **Fault tolerant.** There is no one point of control