

# Midterm Review

*CS 475: Concurrent & Distributed Systems (Fall 2021)*

Yue Cheng

# Midterm

- Thursday, Oct 7<sup>th</sup>, 9:00 – 10:15am
  - 75 minutes
  - Open-book, open-notes (you may use class notes, papers, and lab materials; you may read them on your laptop, **but you are not allowed to use any network**)
- Covering topics from lec-1 to lec-9-2
  - Go-specific questions
  - High-level design questions

# Concurrency in Go

- Labs that were completed
  - Possible race condition bugs in Go
  - Go channels
  - Go mutex locks

`ch := make(ch(int))`  
capacity.  
unbuffered ch  
buffered ch.

# MapReduce + GFS

- Why MapReduce
  - Google workload characteristics
- How MapReduce works
  - Paper
- How data flows within a MapReduce job
  - Use of local file system and use of GFS (Fig. 3 sort app)
- Fault tolerance in MapReduce
  - Backup tasks; task idempotence

# Time & Clocks

Physical time

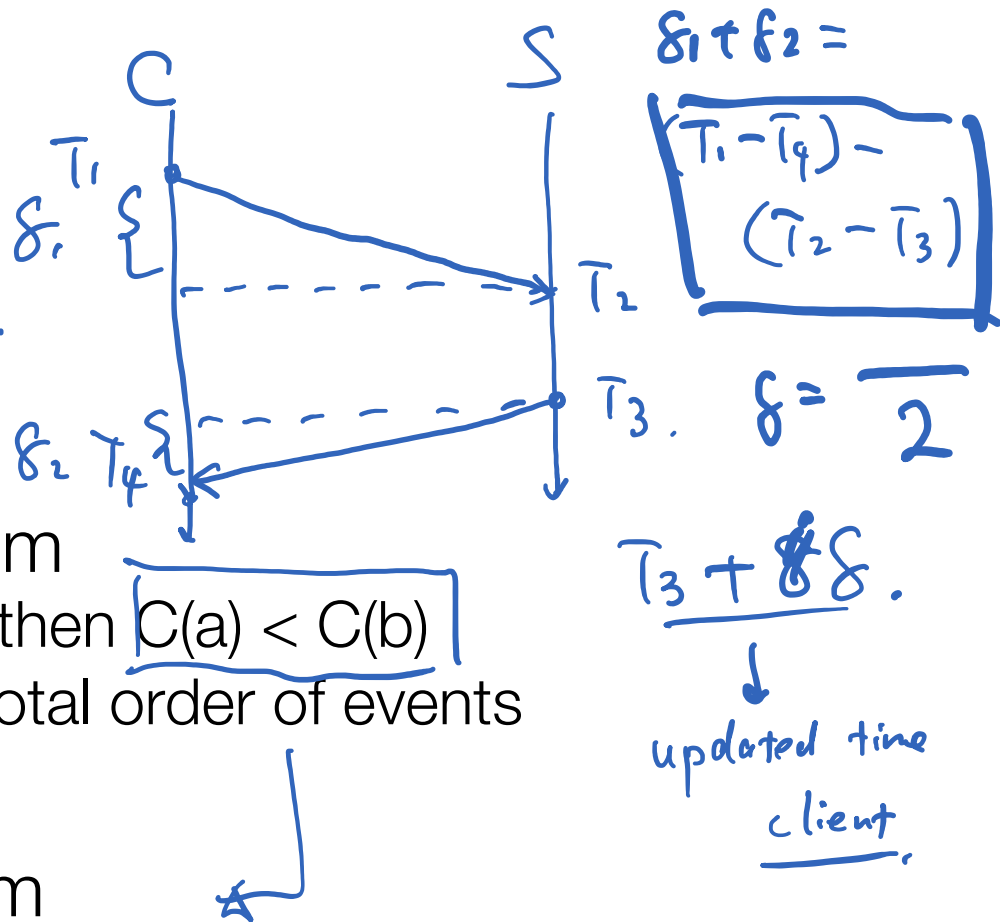
- Cristian's algorithm

- Logical Clock algorithm

- Guarantees if  $a \rightarrow b$ , then  $C(a) < C(b)$
- How to guarantee a total order of events

- Vector Clock algorithm

- If  $V(a) < V(b)$ , then  $a \rightarrow b$
- If  $V(a) \not\leq V(b)$  and  $V(b) \not\leq V(a)$ , then  $a \parallel b$
- Can be used to infer when an event  $b$  was aware of / influenced by  $a$



# Raft

- Raft material: Slides #1-24; paper: Section 1-5

- Safety and liveness requirements

- Election

- Normal operations ←

- • Leader changes

- Rules of selecting the best leader

- • Rules of safe commitment

- Log consistency

Election Timeout

↓  
[T, 2T]

$$7/2 = 3.$$

$$2f + 1 = 7.$$

$$3 + 1 = 4$$

$$f + 1 = 4.$$