Logistics

- Final exam on May 11th
  - Please register on CBTF.
  - Same format and policies as midterms, but longer (3 hours).
    - The exam is closed-book.
    - Only a single double-sided physical cheat-sheet is allowed.
  - **Comprehensive**: includes everything covered in the course.
  - ~50% weightage assigned to materials that were not covered by midterm 1 and midterm 2 syllabus (i.e. blockchains and beyond).
Exam Syllabus

• All topics covered so far
  • Midterm 1 content
  • Midterm 2 content
  • Post-midterm2 content
    • Starting from blockchains, up until distributed datastores.
Exam Syllabus

• Midterm 1 content (included in finals)
  • System model and Failures
  • Failure Detection
  • Clock Synchronization
  • Event ordering and Logical Timestamps
  • Global Snapshot
  • Multicast
Exam Syllabus

- Midterm 2 content (included in finals)
  - Mutual Exclusion
  - Leader Election
  - Consensus
    - Synchronous Consensus
    - Asynchronous Consensus: Paxos, Raft
Exam Syllabus

- Remaining topics (included in finals, ~50% weightage)
  - Blockchains
  - Transaction Processing and Concurrency Control
  - Distributed Transactions
  - External consistency and Spanner
  - Distributed Hash Tables
  - MapReduce
  - Distributed Datastores
Disclaimer

• Quick reminder of the relevant concepts we covered in class post midterm 2.

• Not meant to be an exhaustive review!

• Go over the slides for each class.
  • Refer to lecture videos and textbook to fill in gaps in understanding.
Bitcoin / Blockchains

• How is a new transaction added to the log?
  • How is a block mined, and added to a chain?

• What factors determine the rate at which a block is mined?

• What happens if two nodes mine different versions of a block?

• How is information propagated in a Bitcoin network?

• What guarantees can a Bitcoin network provide?
  • Does it guarantee safety?
  • Does it guarantee liveness?
Transaction Processing

• What are the ACID properties?
  • How is atomicity achieved?
  • What does consistency mean in this context?
  • What does isolation mean, and how is it achieved?
  • What is durability?
Concurrency Control

• What could go wrong if we don’t have isolation?
  • Lost update problem
  • Inconsistent retrieval problem

• What are conflicting operations?

• What is serial equivalence?

• How can we check if an interleaving is serially equivalent?
Concurrency Control

• Pessimistic Concurrency Control
  • Global lock vs per-object locks vs per-object read/write locks
  • Two-phase locking
  • Deadlocks

• Optimistic Concurrency Control
  • Timestamped ordering
Distributed Transactions

• Meeting ACID requirements for distributed transaction:
  • Two-phase commit for atomicity
  • Distributed deadlock detection with two-phase locking.
External Consistency and Spanner

• High-level design only
  • Maintains synchronized clocks and uses physical timestamps.
    • Reads = snapshot at a physical point in time.
  • What is the concurrency control mechanism used by Spanner?
  • Distributed transactions on replicated objects
    • What are the different algorithms / protocols used?
Distributed Hash Tables

• What determines the placement of nodes in a Chord ring with m-bit key space?
• Which node is responsible for storing a given key?
• What are the routing table entries maintained by each node:
  • Finger tables
  • r successor entries
• What is the key lookup protocol in Chord?
• How does Chord handle churns?
  • Stabilization protocol.
MapReduce

- Map: creates intermediate key-value pairs
- Reduce: aggregate by key, and run some computation across all values for the key.
- A MapReduce chain comprises of multiple map-reduce pairs.
- Allows easier parallelization.
  - Multiple map/reduce tasks scheduled in parallel across the servers in a cluster.
- Barrier between a map stage and a reduce stage.
  - No reduce task starts before all map tasks are finished.
Distributed Datastores (Cassandra)

- What is CAP theorem?
  - Can only achieve two out of consistency, availability, and partition-tolerance.
- Cassandra: chooses availability, with *eventual* consistency
  - Key partitioning and replication strategies.
  - How is cluster membership updated?
  - How is a write query executed?
  - How is a read query executed?
  - What are the different consistency levels?
  - What is hinted-handoff and read repair?
Exam Syllabus

• Remaining topics (included in finals, ~50% weightage)
  • Blockchains
  • Transaction Processing and Concurrency Control
  • Distributed Transactions
  • External consistency and Spanner
  • Distributed Hash Tables
  • MapReduce
  • Distributed Datastores
Exam Syllabus

• All topics covered so far
  • Midterm 1 content
  • Midterm 2 content
  • Post-midterm2 content
    • Starting from blockchains, up until distributed datastores.
Good luck!