Summary
Computing is nowadays distributed over several machines, in a local IP-like network, a cloud or a P2P network. Failures are common and computations need to proceed despite partial failures of machines or communication links. This course will study the foundations of reliable distributed computing.

Content
Reliable broadcast
Causal Broadcast
Total Order Broadcast
Consensus
Non-Blocking Atomic Commit
Group Membership, View Synchrony
Terminating Reliable Broadcast
Shared Memory in Message Passing Systems
Byzantine Fault Tolerance
Self Stabilization
Population protocols (models of mobile networks)
Bitcoin, Blockchain
Distributed Machine Learning
Gossip

Keywords
Distributed algorithms, checkpointing, replication, consensus, atomic broadcast, distributed transactions, atomic commitment, 2PC, Machine Learning

Learning Prerequisites
Required courses
Basics of Algorithms, networking and operating systems

Recommended courses
The lecture is orthogonal to the one on concurrent algorithms: it makes a lot of sense to take them in parallel.

Learning Outcomes
By the end of the course, the student must be able to:

• Choose an appropriate abstraction to model a distributed computing problem
• Specify the abstraction
• Present and implement it
• Analyze its complexity
• Prove a distributed algorithm
• Implement a distributed system

Teaching methods
Ex cathedra
Lectures, exercises and practical work

Assessment methods
Final exam (theory)
Project (practice)

Resources

Ressources en bibliothèque
• Introduction to reliable and secure distributed programming / Cachin

Notes/Handbook
Reliable and Secure Distributed Programming
Springer Verlag
C. Cachin, R. Guerraoui, L. Rodrigues

Moodle Link
• https://go.epfl.ch/CS-451