Summary
With the advent of multiprocessors, it becomes crucial to master the underlying algorithmics of concurrency. The objective of this course is to study the foundations of concurrent algorithms and in particular the techniques that enable the construction of robust such algorithms.

Content
Model of a parallel system
A multicore architecture
Processes and objects
Safety and liveness

Parallel programming
Automatic parallelism
Mutual exclusion and locks
Non-blocking data structures

Register Implementations
Safe, regular and atomic registers
Counters General and limited operations
Atomic counters and snapshots

Hierarchy of objects
The FLP impossibility
The consensus number
Universal constructions

Transactional memories
Transactional algorithms
Opacity and obstruction-freedom

Keywords
Concurrency, parallelism, algorithms, data structures

Learning Prerequisites
Required courses
Recommended courses
This course is complementary to the Distributed Algorithms course.

Important concepts to start the course
Processes, threads, data structures

Learning Outcomes
By the end of the course, the student must be able to:
• Reason in a precise manner about concurrency
• Design a concurrent algorithm
• Prove a concurrent algorithm
• Implement a concurrent system

Teaching methods
Lectures, exercises and practical work

Expected student activities
Final exam
Project

Assessment methods
With final exam and project

Resources
Notes/Handbook
Algorithms for Concurrent Systems, R. Guerraoui and P. Kouznetsov

Moodle Link
• https://go.epfl.ch/CS-453