CS-522

Principles of computer systems

Argyraki Katerina, Candea George

Cursus	Sem.	Type
Computer and Communication Sciences		Opt.
Computer science	MA1, MA3	Obl.
Cybersecurity	MA1, MA3	Opt.
SC master EPFL	MA1, MA3	Opt.

Language of teaching	English
Credits	8
Session	Winter
Semester	Fall
Exam	During the semester
Workload	240h
Weeks	14
Hours	**4 weekly**
Courses	4 weekly
Number of positions	

Summary

This advanced graduate course teaches the key design principles underlying successful computer and communication systems, and shows how to solve real problems with ideas, techniques, and algorithms from operating systems, networks, databases, programming languages, and computer architecture.

Content

A modern computer system spans many layers: applications, libraries, operating systems, networks, and hardware devices. Building a good system entails making the right trade-offs (e.g., between performance, durability, and correctness) and understanding emergent behaviors. Great system designers make these trade-offs in a principled fashion, whereas average ones make them by trial-and-error. In this course, we develop a principled framework for computer system design, covering the following topics:

- Modularity and abstraction
- Names
- Layers
- Indirection and virtualization
- Redundancy and fault tolerance
- Replication and consistency
- Laziness and speculation
- Locality and caching
- Hardware/software co-design
- Client/server architectures
- Transactional building blocks

Learning Prerequisites

Required courses

The course is intellectually challenging and fast-paced, and it requires a solid background in operating systems, databases, networking, programming languages, and computer architecture. The basic courses on these topics teach how the elemental parts of modern systems work, and this course picks up where the basic courses leave off. To do well, a student must master the material taught in the following courses:

- CS-202 Computer Systems
- CS-300 Data-Intensive Systems
- CS-302 Parallelism and Concurrency

Recommended courses

The following EPFL courses cover material that significantly helps students taking this course, however they are not required:

- CS-470: Advanced Computer Architecture
- CS-471 Advanced Multiprocessor Architecture
- CS-477 Advanced Operating Systems
- COM-407: Advanced Computer Networking
- CS-438 Decentralized Systems Engineering
- CS-453 Concurrent Computing
- CS-420 Advanced Compiler Construction

Learning Outcomes

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

- Design computer and communication systems that work well
- Analyze how landmark systems apply the course's design principles, and the trade-offs their designers made
- Make rational design trade-offs (e.g., performance vs. correctness, latency vs. availability)
- Anticipate emergent system behaviors (e.g., failure cascades, security vulnerabilities)
- Integrate multiple techniques, ideas, and algorithms from different fields of computing/communication into a working system
- Critique systems research papers and technical documents
- Compose clear technical arguments about system design

Transversal skills

- Demonstrate the capacity for critical thinking
- Access and evaluate appropriate sources of information.
- Summarize an article or a technical report.
- Write a scientific or technical report.
- Manage priorities.

Teaching methods

- A combination of online and in-class lectures
- Interactive discussion sessions
- Reading assignments
- Homework assignments

Expected student activities

- Attend lectures and discussion sessions
- Complete the reading and writing assignments
- Participate actively in the course (physically and online)

Assessment methods

- During the semester (continuous control)
- Final grade is based on specific assignments and quizzes/exam(s)

Supervision

Office hours	Yes
Assistant.e.s	Yes
Forum	Yes

Resources

Virtual desktop infrastructure (VDI)

No

Bibliography

See course website for the latest information and an up-to-date bibliography.

Websites

- https://pocs.epfl.ch