Summary
The course covers fundamental principles for understanding and building systems for managing and analyzing large amounts of data.

Content

Programming methods, including parallel programming:

• Data-parallel programming: Collection abstractions and modern collection libraries.

• Data-flow parallelism vs. message passing. The bulk-synchronous parallel programming model.
• SQL and relational algebra. Expressing advanced problems as queries.

Big data systems design and implementation:
• Scalability. Synchrony. Distributed systems architectures.
• Massively parallel processing operations – joins and sorting
• Challenges of big data machine learning systems.

Changing data:
• Introduction to transaction processing: purpose, anomalies serializability; concurrency
• Commits and consensus.
• Eventual consistency. The CAP theorem. NoSQL and NewSQL systems.

Online / Streaming / Real-time analytics:
• Data stream processing. Windows. Load shedding.
• “Small data”/online aggregation: Sampling and approximating aggregates.
• Incremental and online query processing: incremental view maintenance and materialized views.
Data warehousing: The data warehousing workflow, ETL, OLAP, Data Cubes

Keywords
Databases, data-parallel programming, NoSQL systems, query processing.

Learning Prerequisites
Required courses
CS-322: Introduction to database systems

Recommended courses
CS-323: Introduction to operating systems
CS-206 Parallelism and concurrency

Important concepts to start the course
• Algorithms and data structures – sorting algorithms, balanced trees, graph traversals.
• The Scala programming language will be used throughout the course. Programming experience in this language is strongly recommended.
• Basic knowledge or computer networking and distributed systems

Learning Outcomes
By the end of the course, the student must be able to:
• Choose systems parameters, data layouts, query plans, and application designs for database systems and applications.
• Develop data-parallel analytics programs that make use of modern clusters and cloud offerings to scale up to very large workloads.
• Analyze the trade-offs between various approaches to large-scala data management and analytics, depending on efficiency, scalability, and latency needs
• Choose the most appropriate existing systems architecture and technology for a task

Teaching methods
Ex cathedra; including exercises in class, practice with pen and paper or with a computer, and a project

Expected student activities
During the semester, the students are expected to:
• attend the lectures in order to ask questions and interact with the professor,
• attend the exercises session to solve and discuss exercises,
• solve practical homeworks and/or finish a project during the semester,
• take a midterm
• take a final exam

Assessment methods
Homeworks, written examinations, project. Continuous control

**Supervision**

| Office hours | Yes |
| Assistsants | Yes |
| Forum | Yes |
| Others | Office ours by appointment |

**Resources**

**Bibliography**

Relevant resources (textbook chapters, articles, and videos) posted on moodle page.