

# Computer Networks and Distributed Systems

Programme course

8 credits

Datornät och distribuerade system

TDTS04

Valid from: 2017 Spring semester

**Determined by**

Board of Studies for Computer Science  
and Media Technology

**Date determined**

2017-01-25

## Main field of study

Computer Science and Engineering, Programming

## Course level

First cycle

## Advancement level

G2X

## Course offered for

- Computer Engineering, B Sc in Engineering
- Industrial Engineering and Management - International, M Sc in Engineering
- Industrial Engineering and Management, M Sc in Engineering
- Programming, Bachelor's Programme

## Entry requirements

Note: Admission requirements for non-programme students usually also include admission requirements for the programme and threshold requirements for progression within the programme, or corresponding.

## Prerequisites

Knowledge of C, C++ or Java are required in order to be able to manage the laborations in the course. The student is also assumed to know how to construct and test programs in a Unix/Solaris environment.

## Intended learning outcomes

Computer networks are playing an increasingly important role in the society. It is predicted that there will be 50 billion devices connected to the Internet by 2020. With an enormous increase in the number of wired and wireless devices connected through the Internet, as well as improved network bandwidth and computer capabilities, we are moving towards a society in which users expect to access anything they want, whenever and wherever they are. To enable this trend and all the emerging services provided over the Internet (e.g., social networks, video streaming, and the Internet of things) it is therefore critical for today's computer science students to have a good understanding for computer networks.

TDTS04 is a basic course in computer networks and distributed systems. The course covers the basics for how a network and its applications operate; what a protocol is; how they work, and what the most important applications of the Internet are. It also covers the fundamentals of distributed systems, and some of the design tradeoffs that these systems commonly must take into consideration. In

the labs you will learn more about the mechanisms in some important and fundamental Internet protocols and some basic programming of distributed systems. The final exam will test your understanding and knowledge of the subject. After the course, you are expected to be able to:

- Explain, describe, and analyze a typical network architecture, including the importance of network layers and encapsulation
- Explain the different basic types of protocols, communication channels, and network types

You should have a deep understanding of the network architecture and the protocols associated with the different layers:

- Describe and analyze the most common application architectures in the Internet, how the most important application-layer protocols work and the service they provide
- Analyze and explain important design considerations at the transport layer, including describing how TCP's flow control and congestion control works, and how reliable data transfer is implemented in TCP
- Motivate and explain how routing and forwarding is implemented on the Internet, including describing how IP addressing and fragmentation works
- Describe and explain different link-layer technologies and how they work

You are also expected to understand how distributed systems can be built on-top of the network architecture. More specifically, you should be able to:

- Define what a distributed system is and its most important goals
- Explain the relationship between architectures, processes and communication
- Exemplify different types of transparency, scaling techniques
- Analyze and explain some of the fundamental differences in different system architectures
- Describe and explain how to achieve synchronization, consistency and replication
- Implement, motivate, and explain the design of various types of distributed system architectures, including object-based distributed systems (e.g., using Corba and Java RMI) and Web-based distributed systems (including how a proxy cache works)

## Course content

Basics for distributed systems, definitions, transparency, scaling, architectures, execution techniques for remote procedure calls (rpc and port mapping), and socket-based communication between processes. Pitfalls when implementing distributed systems. Object-based distributed systems (Corba and Java RMI) and web-based distributed systems (HTTP and web services). Communication and process models, naming, synchronization, consistency, caching, replication, and fault tolerance. UTC, NTP. Lamport and vector clocks. Mutual exclusion. Application architectures: client-server (centralized, decentralized/p2p, multi-tiered) and hybrid, exemplified with HTTP, e-mail, file transfer, DNS, Bittorrent, and Skype. CDN and Akamai. Protocol terminology. Protocol layer concept. Reference models for network architectures. System architectures. Application areas for computer networks and distributed systems and examples of commercial network services. Network types, their characteristics and components (router, switch, hub, repeater). Protocol mechanisms and channels. Access network techniques. Different types of MAC protocols. The collision domain concept. The sliding window protocol. Error detection. Local area networks (IEEE 802.3). Wireless networks (Bluetooth, WiFi, WiMax). Extending LANs. Internet and standardization. The TCP/IP protocol family. Distance vector and link-state routing. ICMP. ARP. NAT. DHCP. Naming, addressing and routing in the Internet. TCP's retransmission timer, flow control and congestion control. Reliable data delivery in TCP. Three-way handshake. Network performance issues. Internet applications (the domain name system, e-mail, file transfer, the web, and network management). IP telephony. P2P networks. The DHT data structure. Internet history. Internet design principles. Lan background. Trends.

## Teaching and working methods

The course consists of lectures and laborations.

## Examination

LAB1	Laboratory work	3 credits	U, G
TEN1	Written examination	5 credits	U, 3, 4, 5

## Grades

Four-grade scale, LiU, U, 3, 4, 5

## Other information

*Supplementary courses:*

Advanced Networking; System installation.

## Department

Institutionen för datavetenskap

## Director of Studies or equivalent

Patrick Lambrix

## Examiner

Andrei Gurtov

## Course website and other links

<http://www.ida.liu.se/~TDTS04/>

## Education components

Preliminary scheduled hours: 54 h

Recommended self-study hours: 159 h

## Course literature

Huvudbok: Kurose, J. F. & Ross, K. W. (2012), Computer networking: a top-down approach. Sixth Edition. Pearson.

## Common rules

Regulations (apply to LiU in its entirety)

The university is a government agency whose operations are regulated by legislation and ordinances, which include the Higher Education Act and the Higher Education Ordinance. In addition to legislation and ordinances, operations are subject to several policy documents. The Linköping University rule book collects currently valid decisions of a regulatory nature taken by the university board, the vice-chancellor and faculty/department boards.

LiU's rule book for education at first-cycle and second-cycle levels is available at [http://stydokument.liu.se/Regelsamling/Innehall/Utbildning\\_pa\\_grund-\\_och\\_avancerad\\_niva](http://stydokument.liu.se/Regelsamling/Innehall/Utbildning_pa_grund-_och_avancerad_niva).