CCIS

Master’s Programme in Computer, Communication and Information Sciences

Computer Science Orientation

Petri Vuorimaa

25 Aug 2026

Aalto University
School of Science

CCIS Majors

- Acoustics and Audio Technology
- Communications Engineering
- Computer Science
- Game Design and Development
- Human-computer Interaction
- Machine Learning, Data Science and Artificial Intelligence (Macadamia)
- Signal Processing and Data Science
- Software Engineering
- Speech and Language Technology

Structure of Studies

- **Long major (60 credits):**
  - 30 credits of elective studies
  - elective studies may include an optional minor
- **Compact major (40 credits):**
  - minor (20–25 credits)
  - 25-30 credits of elective studies
- **Altogether major, possible minor and electives have to be 90 credits**
  - thesis is additional 30 credits

CS Tracks

- Algorithms and Theory (Jara Uitto)
- Big Data and Large-Scale Computing (Bo Zhao)
- Secure Systems (Lachlan Gunn)
- Quantum-Aware Software Systems (Alexandru Paler)
- Web Technologies, Applications and Science (Petri Vuorimaa)

Core Courses

- Cloud Software and Systems
- Web Software Development
- Information Security
- Principles of Algorithmic Techniques
- Declarative Programming
- Supervised Machine Learning
- Computational Design and Interaction
- Computer Graphics
- Logic and Hard Computational Problems
- Scalable Systems and Data Management

Rules

- **Select at least five courses from the major core course list**
  - Including the compulsory core course(s) defined by the track
- **Core courses can also be done as part of the Bachelor studies**
  - reduces the number of core course required at the Master level
- **Students who have completed equivalent courses at another university can be excused from taking the core courses with agreement of the professor in charge of the study track**

CS Track Sessions

- Secure Systems -track ***B224*** *CS-Building*
- Algorithms and Theory -track ***A237*** *CS-Building*
- Big Data and Large Scale Computing -track ***B119*** *CS-Building*
- Web Technologies, Applications and Science -track ***A142*** *CS-Building*
- Quantum-Aware Software Systems -track ***A106*** *CS-Building*

Algorithms and Theory Track

Aalto University
School of Science

Faculty

- **Professor in charge: Jara Uitto**
- **Aalto CS faculty members:**
  - Chris Brzuska
  - Corinna Coupette
  - Tommi Junttila
  - Petteri Kaski
  - Barbara Keller
  - Sándor Kisfaludi-Bak
  - Russell W.F. Lai
  - Pekka Orponen
  - Alexandru Paler
  - Jussi Rintanen
  - Jukka Suomela
  - Jara Uitto

HALT Helsinki Algorithms & Theory

algorithms.fi

Research

We are researchers who build the foundations of the modern computerized and networked world. We design and implement fast and reliable algorithms. We explore new kinds of computational settings and study the fundamental limitations of ***what can be computed efficiently****.*

Current activities

Helsinki Algorithms & Theory Days · August 28–29, 2025

HIIT Foundations Friday · monthly, on Fridays at 11.15

HIIT Frontier Friday · monthly, on Fridays at 11.00

Helsinki CS Theory Seminar · weekly, on Wednesdays at 14.15

Helsinki Logic Seminar · weekly, on Wednesdays at 12.00

Aalto University
School of Science

Who?

- Anyone interested in **mathematics** of algorithms and automated reasoning
- Anyone interested in the advanced design, analysis, and applications of **algorithms**
- **Programmers** who *really* want to understand the computer — from individual logic/quantum gates to top-tier supercomputers
- Anyone interested in a **research career** in academia, industry, or public service

Why?

- Curriculum grounded in **leading-edge research** at the department
- Algorithms are part of the critical infrastructure driving **science**, **technology**, and **society**
- Understand the **foundations** and learn to push the **frontiers** of new technologies (e.g. quantum computing, cryptography)
- Fundamentals withstand the **test of time** (e.g. Euclid's algorithm, a key element of modern cryptography, dates back more than 2000 years)

Selected courses from track

CS-E3190	**Principles of Algorithmic Techniques**	**5**	**I–II** *
CS-E3220	**Declarative Programming**	**5**	**I–II** *
CS-E4340	Cryptography	5	I–II *
CS-E4510	Distributed Algorithms	5	I–II *
CS-C3260	Practical Quantum Computing	5	I–II *
CS-E4370	Applied Cryptography	5	III–IV
CS-E4500	Advanced Course in Algorithms	5	III–IV
CS-E4530	Computational Complexity Theory	5	IV–V
CS-E4720	Computational Geometry	5	III–IV
CS-E4800	Artificial Intelligence	5	III–IV
CS-E4595	Competitive Programming	5	I–II *
CS-E4580	Programming Parallel Computers	5	V
CS-E4690	Programming Parallel Supercomputers	5	II
CS-E5485	Algorithms and Society	5	I

*) starts next week

Aalto University
School of Science

Big Data and Large-scale Computing Track

Hong-Linh Truong
Department of Computer Science
linh.truong@aalto.fi, https://rdsea.github.io

Large-scale computing

- **Computing with large-scale ICT infrastructures**
  - on-premise enterprise data centers, high-performance clusters, public cloud data centers, distributed edge systems, and hybrid edge-cloud systems ⇒ heterogeneous computing, memory and networking capabilities
- **Solving problems with a wide range of complex workloads in complex applications and services**
  - long-running, data intensive processing
  - massive and high performance computation
  - large-scale and fast (streaming) ingestion, processing & reasoning
  - complex data workflows for AI-ready data
  - reliable platform services for a very large number of tenants & users

Big/Extreme Data

Data sources

- Social media and business data
- Internet of Things (IoT)/Industrial IoT, Machine-to-Machine (M2M), and manufacturing
- Advanced scientific instruments, e.g., earth observatory, particle accelerators & DNA sequencer
- Personal information and healthcare data
- Open/government data
- Technical documents/writings/texts

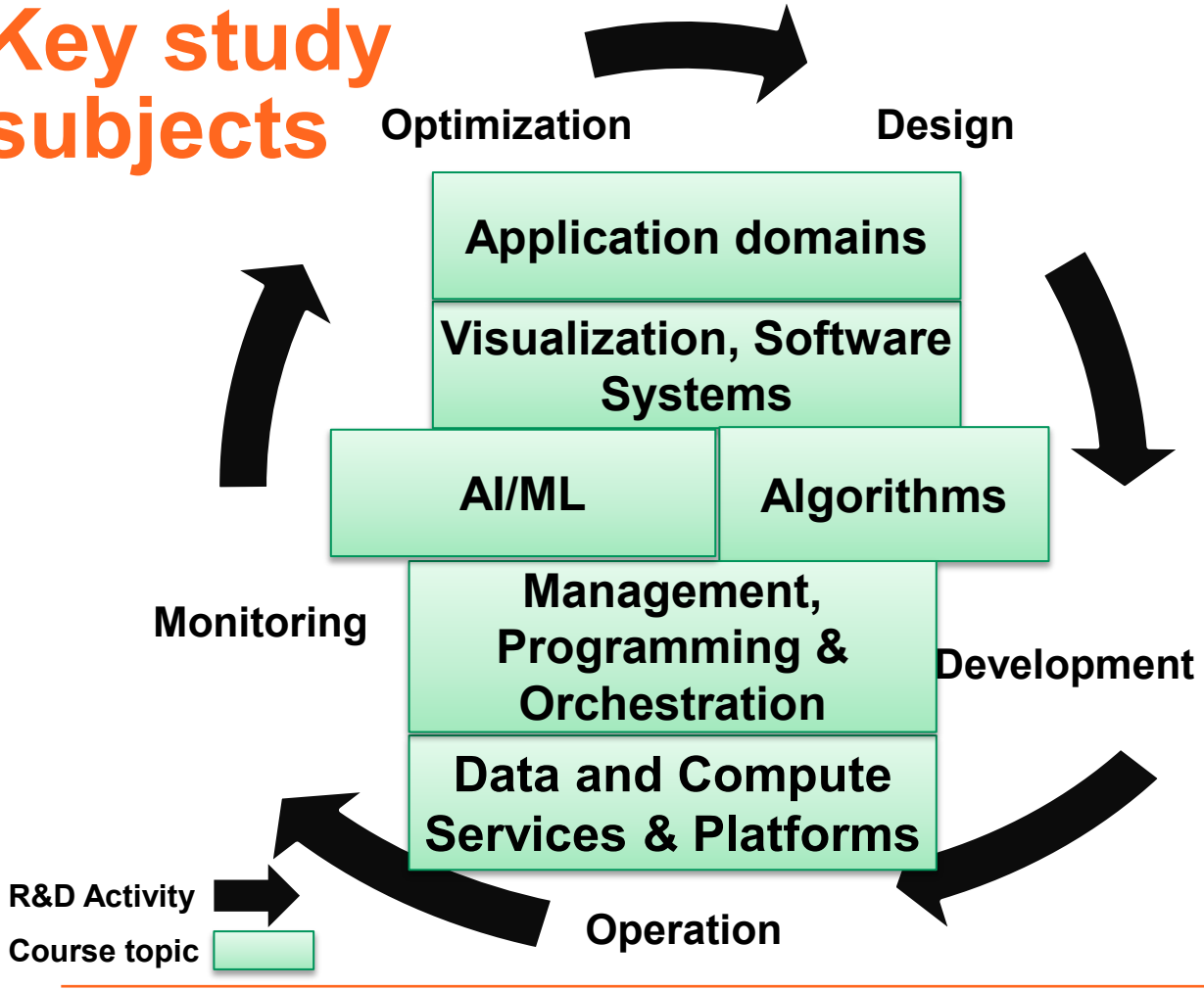
Data characteristics

- Extremely large, complex, dispersed, heterogeneous data from different sources
  - need new techniques/methods
- Diversity and complexity in terms of
  - volume/size (big size, large dataset, massive of small data)
  - complexity (formats, types of data)
  - speed (generating speed, data movement speed)
  - quality, bias, ownerships, and regulations

Big data and large-scale computing track

- **Foundations for data- and compute-intensive services and applications**
  - data systems and data platforms (customers, IoT, IIoT, software, sciences), AI/ML/LLMs systems and services, HPC for sciences (simulation, prediction)
- **Wide range application domains**
  - Business analytics, Healthcare, Software Development, Industry 4.0, Smart City, E-Science, Critical infrastructures operations, Pandemic/Disaster responses, Sustainability, etc.

Key study subjects



Main courses

CS-E4890 Deep Learning
CS-E4500 Advanced Course in Algorithms
CS-E4660 Advanced Topics in Software Systems
CS-E4840 Information Visualization
CS-E4950 Software Architectures
CS-E4770 Designing and Building Scalable Web Applications

CS-E4645 Research Project on Data Intensive Computing
CS-E4650 Methods of Data Mining
CS-E4641 Principles and Techniques of Data Platforms
CS-E4690 Programming Parallel Supercomputers
CS-E4580 Programming Parallel Computers
CS-E4190 Cloud Software and Systems
CS-E4780 Scalable Systems and Data Management

Access to and practice with state-of-the-art infrastructures

- **Computing and data infrastructures**
  - Aalto SCI School's high-performance computing systems
    - *https://scicomp.aalto.fi/triton/*
  - Finland's national supercomputing systems and Clouds
    - *https://research.csc.fi/en/computing*
    - *LUMI (https://www.lumi-supercomputer.eu/)*
  - Also possible with free/university resources from Google Cloud, Microsoft Azure, and Amazon EC2
- **Data**
  - Open and research datasets

Personalizing your study

- **Study subjects in the track can be challenging**
  - complex and challenging software/infrastructure technology stacks
  - engineering emphasis
- **Different emphasis on skills building**
  - e.g., data engineer versus platform architect vs business intelligent analyst
- **Different industrial sectors/academia may influence your learning goals**
  - e.g., manufacturing vs e-health vs enterprise business vs e-science

Secure Systems Track

What is Secure Systems about?

How to protect systems against inside and outside attack

Systems Security

- Networks, software, ships, etc.

Cryptography

- Encryption, authentication, digital signatures, etc.

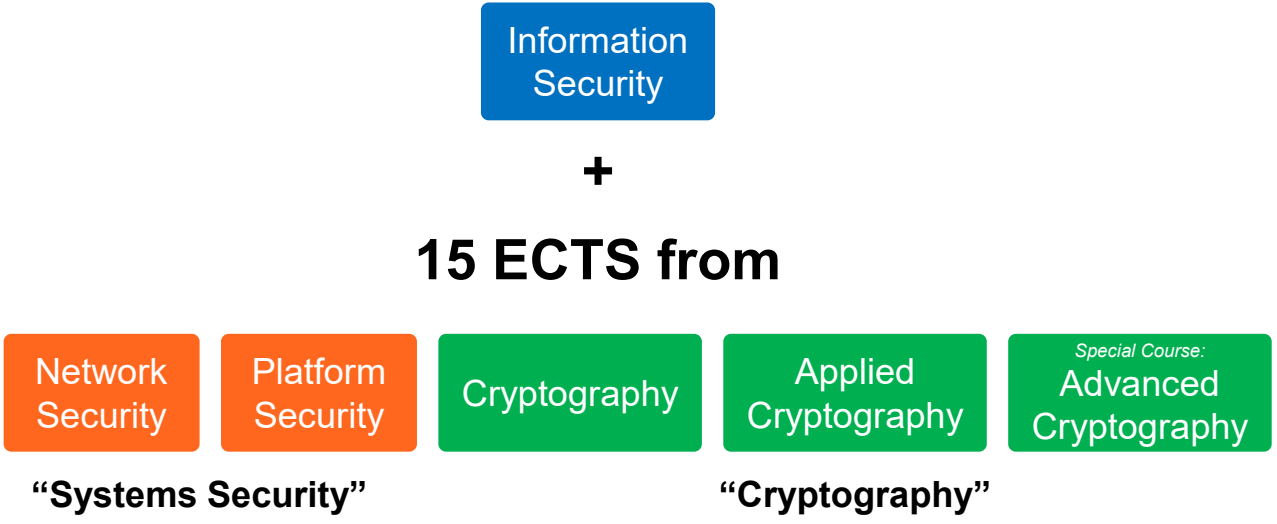
Faculty

Lecturer in charge: Lachlan Gunn

Other key faculty members:

- Tuomas Aura
- Chris Bruszka
- Philip Ginzboorg
- Russell WF. Lai
- Sebastian Szyller

Compulsory Courses



Optional Courses

By agreement masters-level courses in

- computer science
- communications
- mathematics

Recommendations on the right

- Not an exhaustive list!

- Seminar in Computer Science
- Special Course in Information Security
- Cybersecurity Management
- Abstract Algebra
- Number Theory
- Galois Theory
- Probability Theory

Some Recent Thesis Topics

- zk Rollups in Trusted Execution Environments
- Formal Verification of Confidentiality Guarantees in a Blinded Memory SoC
- Agentic AI for Penetration Testing
- Attestation of Distributed Applications
- Utilizing large language models in reverse engineering:
  Annotating decompiled binary code
- Establishing Trusted Channels for Confidential Workloads
- Towards formally verifying the TLS 1.3 Key Schedule Security in SSBee

Quantum-Aware Software Systems Track

Aalto University
School of Science

Software Systems and Technologies

focuses on a wide range of topics, including

- embedded systems
- mobile computing
- virtualization
- cloud computing, and data center networking

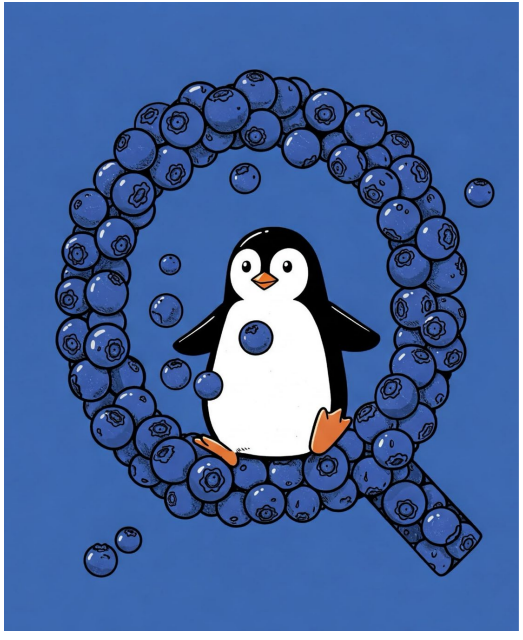
Graduates acquire in-depth knowledge of advanced software systems and architectures, preparing them for emerging platforms and technologies.

Key Course Areas

- **Algorithms and Data Structures**: Start with a strong theoretical base. Look for courses that cover algorithms designed for parallel architectures and the analysis of their computational complexity. This provides the fundamental knowledge needed to write efficient, scalable code.
- **High-Performance Computing (HPC)**: This is the core of parallel computing. Courses in this area will teach you how to program modern multi-core processors, GPUs, and distributed supercomputers like LUMI. You'll learn concepts such as thread-level parallelism, message passing (e.g., MPI), and shared memory programming (e.g., OpenMP).

Key Course Areas

- **Cloud and Distributed Systems**: Modern parallel computing often takes place in the cloud. Courses in this area will help you understand the architectural principles of distributed services, virtualization, and cloud platforms. You'll learn to develop and deploy large-scale, parallel applications that can leverage cloud resources.
- **Quantum Computing**: For a more forward-looking approach, you can explore the emerging field of quantum computing. The program offers research opportunities and courses that provide an introduction to the unique challenges and paradigms of programming quantum computers.



Web Track

Faculty

- **Professor in charge: Petri Vuorimaa**
- **Other key faculty members:**
  - Senior University Lecturer Arto Hellas
  - University Teacher Juho Vepsäläinen (fixed term)

Compulsory Courses

- Design of WWW Services I-II/1st year
- Web Software Development (core course) II/1st year
- WWW Applications I-II/2nd year

Optional Courses

- Designing and Building Scalable Web Applications
- Cross-Platform Development
- Scalable Systems and Data Management
- Multimedia Systems
- User Interface Construction
- Semantic Web
- Special Assignment in Computer Science
- Artificial Intelligence
- Complex Networks
- Principles and Techniques of Data Platforms
- Information Visualization
- Research Methods Foundations

Some Recent Thesis Topics

- Artificial Intelligence for Web Development
- Leveraging Ensemble Learning with Web Technologies for Adaptive News Recommendations
- Reviewing E-commerce Accessibility
- Candidate Participation in Voting Advice Applications
- Enhancing Web Accessibility: Developer Perspectives and Practical Solutions
- Improving and Automating Design System Testing
- Kubernetes Networking: Comparative Insights into API Gateways and Service Mesh Implementations