Università
della
Svizzera
italiana

**Faculty
of
Informatics**

Plan of studies 3-5-8

2018/
19

2018/19

Plan of studies 3-5-8

2018/19

Index

Preface

Preface

Informatics is information plus automation. It covers techniques and methods to represent, organize, store, access, communicate, and process information. Informatics is a bit like mathematics. It is a universal language and a powerful formalism to describe and analyze, and it is fundamental for science and engineering. Informatics is a bit like engineering. It is the practical and clever application of principles for a myriad of good uses. It is invention, innovation, technology, and design. Informatics is a bit like art, too. It is a mix of imagination and skills, and also a sense of beauty. And it is absolutely fascinating!

Informatics is everywhere. It impacts and contributes to all aspects of human life in modern societies, and therefore it is a platform for exciting careers, not only in information technology but also in economics, health, aerospace, entertainment, and so many other sectors.

Informatics is our passion. The Faculty of Informatics is home to a diverse group of excellent researchers and dedicated teachers. We are engaged in several national and international research projects, and we offer a full curriculum that includes Bachelor, Master, and PhD programmes, all taught in English. The Faculty continues to grow while keeping an enthusiastic, exciting, and vibrant environment for students and researchers.

Prof. Antonio Carzaniga
Dean of the Faculty of Informatics

Faculty of Informatics

The Faculty of Informatics

Established in October 2004, USI's Faculty of Informatics is dedicated to high quality teaching and research. The mission of the Faculty is to conduct research and produce results in the field of informatics and to equip students with creative problem-solving skills that enable them to address complex problems in business and society.

The Faculty features 8 main areas of research, namely: Software Engineering, Computer Systems, Computational Science, Geometric and Visual Computing, Information Systems, Intelligent Systems, Programming Languages, and Theory and Algorithms.
Born as a traditionally flat structure, the Faculty also features some institutes, such as the Advanced Learning and Research Institute (ALaRI), the Institute of Computational Science (ICS), and the Software Institute (SI), and as partner institute the Dalle Molle Institute for Artificial Intelligence (IDSIA) and the Swiss National Supercomputing Centre (CSCS).

Teaching excellence is assured by an international faculty, low student/academic staff ratio and a modern, innovative curriculum.
The tuition language in the Faculty is English.
The undergraduate programme is project-based and comprises six semesters of highly integrated courses and team-oriented projects.
For graduate students, the Faculty offers several specialized Master's programmes (also in cooperation with the Faculty of Economics) and a research-oriented PhD programme. The PhD programme is highly selective and gives students the opportunity to participate in national and international research projects.

The Faculty has an active network of research partnerships with other Swiss and international centres. The national and international networks support research collaborations and student mobility.

Executive bodies

Dean
Prof. Antonio Carzaniga
office Informatics Building, office 218
tel +41 (0)58 666 46 89
e-mail dean.inf@usi.ch

Vice-Dean
Prof. Marc Langheinrich
office Informatics Building, office 106
tel +41 (0)58 666 43 04
e-mail marc.langheinrich@usi.ch

Vice-Dean
Prof. Fernando Pedone
office Informatics Building, office 217
tel +41 (0)58 666 46 95
e-mail fernando.pedone@usi.ch

The Dean and Vice-Deans
are available for meetings by appointment.

Dean's Office secretaries
Elisa Larghi,
Janine Caggiano,
Nadia Ruggiero,
Jacinta Vigini
office +Informatics Building, office 120
tel +41 (0)58 666 46 90
fax 41 (0)58 666 45 36
e-mail decanato.inf@usi.ch
bookings.inf@usi.ch
Office hours 09.15-12.30

Coordinator of Faculty
activities/projects and external relations:
office Ing. Mauro Prevostini
tel Via Balestra Building, office 301
e-mail +41 (0)58 666 47 17
mauro.prevostini@usi.ch

Mobility Delegate
office Prof. Natasha Sharygina
tel Informatics Building, office 220
e-mail +41 (0)58 666 42 99
natasha.sharygina@usi.ch

Faculty IT office:
Ing. Giacomo Toffetti-Carughi

Bachelor's programme Director
office Prof. Laura Pozzi
tel Informatics Building, office 206
e-mail +41 (0)58 666 43 01
bachelor.dir.inf@usi.ch

Master in Informatics
office Prof. Kai Hormann
tel Informatics Building, office 105
e-mail +41 (0)58 666 43 27
kai.hormann@usi.ch

office Prof. Evanthia Papadopoulou
tel Informatics Building, office 209
e-mail +41 (0)58 666 41 22
evanthia.papadopoulou@usi.ch

Master in Artificial Intelligence
office Prof. Luca Maria Gambardella
e-mail IDSIA, Manno
luca.maria.gambardella@usi.ch

office Prof. Jürgen Schmidhuber
e-mail IDSIA, Manno
jurgen.schmidhuber@usi.ch

Master in Computational Science
office Prof. Olaf Schenk
tel LAB Glass Building, office 201
e-mail +41 (0)58 666 48 50
olaf.schenk@usi.ch

Master in Cyber-Physical and Embedded Systems
Prof. Cesare Alippi
office Informatics Building, office 114
tel +41 (0)58 666 43 39
e-mail cesare.alippi@usi.ch

Master in Financial Technology and Computing
Prof. Marc Langheinrich
office Informatics Building, office 106
tel +41 (0)58 666 43 04
e-mail marc.langheinrich@usi.ch

Prof. Fernando Pedone
office Informatics Building, Office 217
tel +41 (0)58 666 46 95
e-mail fernando.pedone@usi.ch

Master in Management & Informatics
Prof. Marc Langheinrich
office Informatics Building, office 106
tel +41 (0)58 666 43 04
e-mail marc.langheinrich@usi.ch

Master in Software & Data Engineering
Prof. Cesare Pautasso
office Via Balestra, Office 305
tel +41 (0) 58 666 43 11
e-mail cesare.pautasso@usi.ch

Prof. Gabriele Bavota
office Via Balestra, office 201
tel +41 (0)58 666 49 28
e-mail gabriele.bavota@usi.ch

PhD programme Director
Prof. Walter Binder
office Office SI-204
tel +41 (0)58 666 43 03
e-mail phd.dir.inf@usi.ch

Prof. Olaf Schenk
office LAB Glass Building, office 201
tel +41 (0)58 666 48 50
e-mail phd.dir.inf@usi.ch

The programme directors and delegates are available by appointment.

Faculty's governing bodies

The Faculty's governing bodies include: the Faculty Council, the Professors Council, and the Dean's Office.

Faculty Council

The highest body of the Faculty is the Faculty Council.
It comprises:

- all tenured professors (full and associate), the assistant professors and adjunct professors of the Faculty;
- one teacher representative (with one- or two-year contract),
- one post-doctoral researcher representative, one PhD student representative and one student representative (Bachelor and Master).

Full professors

Cesare Alippi
Walter Binder
Michael Bronstein
Antonio Carzaniga
Fabio Crestani
Patrick Eugster
Illia Horenko
Kai Hormann
Rolf Krause
Marc Langheinrich
Michele Lanza
Evanthia Papadopoulou
Michele Parrinello
Cesare Pautasso
Fernando Pedone
Mauro Pezzè
Laura Pozzi
Olaf Schenk
Jürgen Schmidhuber
Natasha Sharygina
Paolo Tonella
Ernst Wit
Stefan Wolf

Associate professors

Carlo A. Furia
Matthias Hauswirth
Igor Pivkin
Silvia Santini

Assistant professors

Gabriele Bavota
Piotr Didyk
Michael Multerer
Nate Nystrom
Robert Soulé

Adjunct professors

Luca Maria Gambardella

Faculty Representatives

- Students
  Aron Fiechter (Valerie Burgener)
- PhDs
  Daniele Grattarola (Ioannis Mantas)
- Post-docs
  Tbc
- Teachers
  Marco Brambilla

Professors Council

The Professors Council is made up of all tenured professors (full and associate) of the Faculty.

Dean

The current Dean is Prof. Antonio Carzaniga.

For the specific duties of each body please refer to the Statute of the Faculty.

Research institutes

ALaRI Advanced Learning and Research Institute

ALaRI, established in 1999, is the Advanced Learning and research Institute at the Faculty of Informatics at the Università della Svizzera italiana in Lugano, Switzerland. ALaRI's mission is promoting research and education in Cyber- Physical and embedded Systems. Aware of the real need for an interdisciplinary approach to education, ALaRI equips students with a unique body of knowledge ranging from electronics to informatics, from sensors and actuators to communication, from physical modeling to application design including interpersonal skills, indispensable in today´s industry, such as team work, complex-project management, and market sensitivity. The research activities focus on topics of great scientific interest and industrial applicability, based on real-life design issues such as physical modeling, highlevel system design, Internet of Things, smart grids, wireless communication as well as system properties such as performance, dependability, intelligence, security and real time

Director of ALaRI

- Prof. Cesare Alippi
  Informatics Building
  via Giuseppe Buffi / 6900Lugano
  +41 (0)58 666 46 90
  master@alari.ch
  www.alari.ch

ICS Institute of Computational Science

Advanced mathematical modeling and High-Performance methods in numerical simulations open new perspectives for science, research and economy. Exploiting the capabilities of modern supercomputers, increasingly complex problems can be tackled - covering a very broad spectrum of disciplines, from exact and natural sciences to economics and social sciences, including biomedical, environmental, materials, and engineering sciences. The ICS provides a unique research environment, where strong competences in modeling, simulation and information science come together in an open and application oriented atmosphere.
Our aim is the efficient modeling and simulation of nonlinear processes on multiple scales in scientific and biomechanical applications. Current projects include biomechanics, contact

problems in elasticity with and without friction, nonconforming domain decomposition methods, nonlinear and non-smooth multigrid methods, parallel nonlinear solution methods, adaptive finite elements for complex geometries, and the coupling of molecular dynamics and finite element discretizations.

Director of ICS

- Prof. Rolf Krause
  LAB Glass Building
  via Giuseppe Buffi / 6900Lugano
  +41 (0)58 666 46 90
  ics@usi.ch
  www.ics.usi.ch

SI Software Institute

The Software Institute (SI) is part of the Faculty of informatics of the Università della Svizzera italiana (USI), located in beautiful Lugano, in Southern Switzerland. At the SI, our strength is discovering, designing, and developing new ideas that ease the conception of modern software systems. Our research is rooted both in sound theoretical models as well as practical, real-life questions that impact modern society, a society where reliable, well engineered software systems have become quintessential. The SI is a center of excellence committed to the teaching, the research and the development of software. The SI is directed by Michele Lanza and features renowned software researchers among its members: Profs. Gabriele Bavota (Software Analytics & Empirical Software Engineering), Matthias Hauswirth (Software Performance), Cesare Pautasso (Software Architecture & Web Engineering), Carlo Alberto Furia (Software Engineering, Formal Methods & Verification), and Paolo Tonella (Software Testing).

Director of SI

- Prof. Michele Lanza
  Via Balestra / 6900Lugano
  +41 (0)58 666 46 90
  decanato.inf@usi.ch
  www.si.usi.ch

Partner institutes

IDSIA Istituto Dalle Molle di Studi sull'Intelligenza Artificiale

IDSIA was founded in Lugano in 1988 by Angelo Dalle Molle (1908-2002), an Italian philanthropist whose vision was a world where technological progress and human development could both contribute to the improvement of our quality of life.
Dalle Molle was a precursor of electric mobility, and he estabilshed a Trustee in Switzerland to promote creative scientific research, free from the bureaucratic ties of university institutions. Nowadays the institutes founded by Angelo (IDSIA in Lugano, IDIAP in Martigny, and ISSCo in Geneva) are integrated in the local institutions.
Since the foundation of USI and SUPSI in Canton Ticino, IDSIA has been designated to be a "bridge" between these two institutions. For this reason IDSIA activities span from fundamental to applied research, transferring its knowledge into applications in the real world.

Director of IDSIA

- Prof. Luca Maria Gambardella
  +41 (0)58 666 66 60
  IDSIA, Galleria 2, 6928 Manno
  luca@idsia.ch
  www.idsia.ch

CSCS Swiss National Supercomputing Centre

Founded in 1991, CSCS develops and provides the key supercomputing capabilities required to solve challenging problems in science and/or society. The centre enables world-class research with a scientific user lab that is available to domestic and international researchers through a transparent, peer-reviewed allocation process. CSCS's resources are open to academia, and are available as well to users from industry and the business sector. The centre is operated by ETH Zurich and is located in Lugano. CSCS and the Università della Svizzera italiana coordinate the Swiss Platform for Advanced Scientific Computing (PASC); a joint effort of all Swiss universities to create a long-term research-driven cooperation network in computational science. The PASC overarching goal is to position Swiss computational sciences in the emerging exascale-era and aims to provide the Swiss scientific community with the tools to make the best use of the new generations of supercomputing machines to solve key problems for science and society. It addresses important scientific research issues in high-performance computing and computational science in different domain sciences through interdisciplinary collaborations between domain scientists, computational scientists, software developers, computing centres and hardware developers.

Director of CSCS

- Prof. Thomas C. Schulthess
  CSCS, Via Trevano 131, 6900 Lugano
  +41 (0)91 610 82 11
  +41 (0)91 610 82 82
  info@cscs.ch
  www.cscs.ch

Rector, Administration and Services

Rector
Prof. Boas Erez

Administration and Services Lugano Campus

USI administration comprises of different services and it is under the Rectorate responsibility, specifically under the responsibility of the Secretary General and the Administrative Director, and through the Rectorate, under the University Council.

Administrative Director
Cristina Largader

Secretary General
Albino Zgraggen

Human Resources
personale.lu@usi.ch

Front desk

office	Main Building, Office 220
tel	+41 (0)58 666 40 00
web	www.attestati.lu.usi.ch

Accommodation

office	Main Building, Office 202
tel	+41 (0)58 666 44 89
e-mail	alloggi@usi.ch
web	www.usi.ch/en/alloggi

Alumni

office	Main Building, Office 303
tel	+41(0)58 666 46 06
e-mail	alumni@usi.ch
web	www.usi.ch/en/alumni

Career

office	Main Building, Office 303
tel	+41(0)58 666 46 06
e-mail	careerservice@usi.ch
web	www.usi.ch/en/career

Equal Opportunities

office	Main Building, Office 257
tel	+41 (0)58 666 46 12
e-mail	equality@usi.ch
web	www.usi.ch/equality

Financial Controlling

office	Main Building, Office 221

International Relations and Study-abroad Office

office	Main Building, Office 202
tel	+41 (0)58 666 46 26
e-mail	relint@usi.ch
tel	www.usi.ch/en/relint

IT

office	Main Building, Office 164
tel	+41 (0)58 666 46 10
e-mail	sistema.lu@usi.ch
web	www.usilu.net

Legal

office	Lambertenghi Building
tel	+41 (0)58 666 46 16
e-mail	serviziogiuridico@usi.ch

Library

tel	+41 (0)58 666 45 09
e-mail	library.lu@usi.ch
web	www.bul.sbu.usi.ch

Logistics

office	Main Building, Office 169
e-mail	logistica.lu@usi.ch

Campus Security

office	Main Building, Office 169
tel	+41 (0)58 666 47 30
e-mail	sicurezza@usi.ch

Media and Communication

office	Lambertenghi Building
tel	+41(0)58 666 47 92
e-mail	press@usi.ch
web	www.usi.ch/en/media-and-communications

Psychological Counselling

e-mail	info@psicologi-ticino.ch
web	www.psicologi-ticino.ch

Quality Assurance

office	Lambertenghi Building
tel	+41 (0)58 666 41 99
e-mail	quality@usi.ch

Research

office	Lambertenghi Building
tel	+41 (0)58 666 46 14
e-mail	servizio.ricerca@usi.ch
web	www.ticinoricerca.ch

Sport

office	Main Building, Office 135
tel	+41 (0)58 666 47 97
e-mail	sport@usi.ch
web	www.sport.usi.ch

Study Advisory and Promotion

office	Main Building, Office 303 (Level 3)
tel	+41 (0)58 666 47 95
web	www.usi.ch/it/contattaci

Web and Graphic Design

office	Main Building, Office 401
tel	+41 (0)58 666 45 15
e-mail	web@usi.ch
tel	+41 (0)58 666 45 17
e-mail	grafica@usi.ch

Start-up Promotion Center

office	Maderno Building
tel	+41 (0)58 666 41 64
e-mail	admin@cpstartup.ch
web	www.cpstartup.ch

E-learning (eLab)

office	Main Building, office 139
tel	+41 (0)58 666 47 60
e-mail	info@elearninglab.org
web	www.elearninglab.org/en/

L'ideatorio

office	Lambertenghi Building
tel	+41 (0)58 666 45 20
web	www.ideatorio.usi.ch/

Development and Fund Raising

office	Main Building, Office 401
tel	+41 (0)58 666 49 27

Copy Center Onys

office	Main Building, Office 121
tel	+41 (0)58 666 45 86
e-mail	copycenter@usi.ch

USI online services and resources

- **Student platform**
  www.usilu.net

- **Faculty Course registration platform**
  http://teaching.inf.usi.ch

- **eCourses platform (Moodle)**
  www.icorsi.ch

- **Exam registration and consultation:**
  www.esami.lu.usi.ch

- **Email access**
  www.mail.usi.ch

- **FTP access**
  ftp.lu.usi.ch

- **Corporate design**
  www.usi.ch/en/images-and-logotype

Student associations

Several student associations have been created within the University. The main objectives are to improve relations between students and the institution and to enrich the range of educational and recreational offer during school. The associations are concerned mainly with the collection of didactic material, organisation of parties and meetings, cultural and sporting events, and networking among University students and the business world.
www.desk.usi.ch/en/list-acknowledged-student-associations

More information:
www.usi.ch/en/administration-and-services

Academic calendar

Academic calendar

The academic year goes from September until June. Courses are held from September until December and from February until June. The semester includes 3 exam sessions (January, June and September).

Autumn Semester 2018-19	Formal registration	3 - 21 September 2018
	Classes begin	17 September 2018
	Classes end	21 December 2018
Winter Exams	Registration	19 November - 3 December 2018
	Exams session	14 - 25 January 2019
	Master defenses	28 January - 1 February 2019
Spring Semester 2019	Classes begin	18 February 2019
	Classes end	31 May 2019
Summer Exams	Registration	18 April - 2 May 2019
	Exams session	11 - 21 June 2019
	Master defenses	24 - 28 June 2019
Autumn Exams (recovery)	Registration	22 July - 5 August 2019
	Exams session	2-13 September 2019
No classes:	All Saint's Day	1 November 2018
	St. Joseph Day	19 March 2019
	Easter holidays	19 - 28 April 2019
	Labour Day	1 May 2019
	Corpus Christi	31 May 2019

Bachelor in informatics

BSc

Bachelor’s degree programme

Introduction

The Bachelor of Science in Informatics introduces students to the theory and practice of informatics. It emphasizes theoretical foundations, technology, systems thinking, and soft skills like communication and teamwork. The curriculum is structured around four areas of learning essential for a truly interdisciplinary education:

Theory

The principles and foundations were established in the 20th century. These foundations help the students understand the potential and limits of computing. The theoretical subjects represent a solid basis to conduct sound scientific analysis and design.

Technology

Informatics is in permanent and fast-paced evolution, characterized by rapid changes in technology. Students are exposed to the most recent technological advances and learn to cope with technological change and evolution, as well as the impact of technology on society.

Systems thinking

Informatics systems today form the foundations of many societal, governmental, and business systems and services. Students learn to view a computer-based system as a component of a larger environment rather than an isolated system.
Communication and Teamwork. Information technology projects are intrinsically interdisciplinary. Informatics professionals work in teams to identify complex problems and develop appropriate solutions. Students learn to communicate, to work with others in teams, and to present the results of their work. The program is based on the european Credit Transfer System (ECTS), which is recognized by all universities in Europe. The three-year Bachelor degree (BSc) is followed by a two-year graduate study programme, leading to a Master degree (MSc). The Faculty offers six Master programmes (see p. 83).

Mobility

A student can take part in a mobility or student exchange programme and undertake a semester in another university for a maximum of 30 ECTS in one semester. The student must discuss the choice of host institution and the study plan with the Bachelor director and obtain approval. The mobility period generally lasts one semester; it may be extended, subject to approval of the Bachelor director, to a maximum of two consecutive semesters. For all information about mobility please consults the International Relations and Study-abroad Office at http://www.usi.ch/en/relint.

Study plan

The Bachelor programme consists of an innovative, project-based, team-oriented curriculum of six semesters (three years) and corresponds to 180 ECTS credits proportionally distributed (30 ECTS for each semester). In the first four semesters, students work on group projects. In the fifth semester students are required to do an internship in industry. In the sixth semester, they work on an individual final project in which they use all the acquired knowledge to solve an interesting problem. The Bachelor students have opportunities for summer internships both at companies and at the university.

Study programme
Bachelor curriculum 2018-2019

	Course	Instructor	ETCS
First semester 30 ETCS	Calculus	Kai Hormann	6
	Computer Architecture	M. Langheinrich, S. Santini	6
	Programming Fundamentals 1	Nate Nystrom	9
	Technical English	Jim Kaufmann	3
	Software Atelier 1: Fundamentals of Informatics	Gabriele Bavota	6
Second semester 30 ETCS	Algorithms & Data Structures	Antonio Carzaniga	6
	Discrete Structures	Stefan Wolf	6
	Linear Algebra	Igor Pivkin	6
	Programming Fundamentals 2	Matthias Hauswirth	6
	Software Atelier 2: Human-Computer Interaction	Monica Landoni	6
Third semester 30 ETCS	Automata & Formal Languages	Laura Pozzi	3
	Computer Networking	A. Carzaniga, S. Santini	6
	Probability & Statistics	Davide Eynard	6
	Programming Fundamentals 3	Walter Binder	6
	Software Atelier 3: The Web	Cesare Pautasso	9
Fourth semester 30 ETCS	Data Management	Patrick Eugster	6
	Introduction to Computational Science	Michael Multerer	3
	Operating Systems	Fernando Pedone	6
	Systems Programming	Antonio Carzaniga	6
	Software Atelier 4: Software Engineering Project	A. Mocci, L. Ponzanelli	9
Fifth semester 30 ETCS	Algorithms & Data Structures 2	Evanthia Papadopoulou	3
	Artificial Intelligence	Luca Maria Gambardella	3
	Computer Graphics	K. Hormann, P. Didyk	6
	Experimentation & Evaluation	M. Hauswirth, M. Langheinrich	3
	Information Retrieval	Fabio Crestani	6
	Numerical Computing	Olaf Schenk	6
	Software Atelier 5: Field Project	M. Lanza, M. Prevostini	9
Sixth semester 30 ETCS	Languages & Compilers	Nate Nystrom	6
	Machine Learning	Cesare Alippi	6
	Optimization Methods	R. Krause, M. Nestola	6
	Theory of Computation	Natasha Sharygina	6
	Bachelor Project	Mauro Pezzè	12

Bachelor of Science

First year

Bachelor of Science **First year**
Fall semester

Lecture
6 ECTS

Calculus

Instructor Kai Hormann

Description This course teaches the essentials from real analysis, which are relevant to informatics. It consists of five chapters. After revisiting basic facts about natural numbers, integers, and rational numbers, the first milestone is to understand the concept of real numbers and their properties. We then study sequences and series of real numbers and learn about the idea of convergence. The third chapter introduces real functions in one variable and focuses on the property of continuity and its consequences. Differentiation and integration are covered in the last two chapters. After finishing this course, students possess the mathematical skills required for solving basic problems in a formal and structured way and they will have developed a good understanding of differential and integral calculus. Whenever possible, applications of theoretical concepts are highlighted and part of the homework assignments deal with the implementation of numerical algorithms to practically experience the mathematical concepts.

References

- Kenneth A. Ross. Elementary Analysis: The Theory of Calculus. Undergraduate Texts in Mathematics. Springer, 1980
- Additional material will be provided through the course homepage.

Lecture
6 ECTS

Computer Architecture

Instructors Marc Langheinrich, Silvia Santini

Description The class teaches the basic principles of how a computer functions, from the very basic building blocks (transistors and logical gates) to the more complex components (CPU, memory, buses, I/O interfaces). Students learn how one can describe the basic operations in a computer using digital logic, and how these operations can be realized in both hardware and software. Students gradually combine these basic operations into a "microarchitecture" -- a softwarecontrolled datapath that connects digital memory with an arithmetic-logical unit -- on which one can then build more and more complex "layers" that will finally allow the writing of complex programs in human-readable programming languages. This knowledge not only forms the basis for understanding how something as complex as a modern computer actually works, but is also a pre-requisite for learning about many advanced topics in informatics, such as Hardware/Software Co-Design, System Programming, Compilers, and Operating Systems.

References

- "Structured Computer Organisation", Andrew S. Tanenbaum, Todd Austin. 6th Edition (International), Pearson Education, 2012, ISBN-10: 0273769243, ISBN-13: 978-0273769248

Lecture & Lab
9 ECTS

Programming Fundamentals 1

Instructor Nate Nystrom

Description PF1 is a first course in programming -- applying computation to problem solving. The course is aimed at students with little or no prior programming experience. We'll be using a programming language called Racket, in which we will practice functional programming. But, this is not a course about Racket; it's a course about software construction: designing programs and then translating designs into implementations. Designing software means making wise choices about data structures, algorithms, and program organization. Implementing means more than just writing code: it means making wise decisions about systems and interfaces. By the end of the course, students should be familiar with various programming constructs universal to all languages, they should be able to analyze problems and then create programs to solve them.

References

- "How to Design Programs", Second Edition, Matthias Felleisen, Robert Bruce Findler, Matthew Flatt, Shriram Krishnamurthi. 2018. http://htdp.org/2018-01-06/Book/

Atélier
6 ECTS

Software Atelier 1: Fundamentals of Informatics

Instructor Gabriele Bavota

Description The first of the ateliers, which are a crucial part of our Bachelor curriculum is roughly divided into three main pieces.
On the one hand the students will obtain first-hand experience with a variety of tools of the trade, such as LaTex, HTML, Versioning (SVN), and the shell. Second, the students will get an overview of the history of computer science since its very beginning up to the present day. The third part of the atelier is dedicated to the semester project that the students will do as part of the Programming Fundamentals course.

Lecture
3 ECTS

Technical English

Instructor Jim Kauffman

Description This course focuses on improving the four language skills: listening, reading, writing and speaking. In addition, emphasis is given to understanding and using different grammatical structures, as well as expanding and applying vocabulary. Course content includes a variety of technical areas, for example technology, planning, projects, design and careers. Information is provided on how to obtain international language qualifications through either the Cambridge English Language Assessment exams (for example, the First Certificate Exam) or the International English Language Testing System (IELTS). If participants are interested in pursuing these qualifications, some preparation and practice for the exams will be included in the course.

References

- For the Intensive Course: New Language Leader Upper Intermediate, published by Pearson, ISBN 978-1-4479-6154-3
- For the Technical English Course: Technical English 4, published by Pearson, ISBN 978-1-4082-2955-2

Lecture
6 ECTS

Algorithms & Data Structures

Instructor Antonio Carzaniga

Description

Algorithms and data structures are fundamental to computer science. They are the essence of computer programs.
The performance of any software system depends on the efficiency of its algorithms and data structures, and more generally, the study of algorithms provides insights into the nature of problems. This course provides students with the basic knowledge and skills necessary to design and reason about algorithms, and to understand the purpose as well as the strengths and weaknesses of some of the most fundamental algorithms and data structures. The course covers basic notions of: complexity, asymptotic worst-case and average complexity, big-O notation, complexity classes; general algorithmic strategies, brute force, greedy, divide-and-conquer, and dynamic programming; common algorithms, searching and sorting, elementary graph algorithms, string matching; basic data structures, stacks, queues, linked lists, rooted trees; more advanced data structures, B-trees, heaps, hash tables.

Recommended courses

- Programming Fundamentals 1

References

- Textbook: "Introduction to Algorithms" (Third Edition), by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Cliff Stein, published by MIT Press and McGraw-Hill.

Lecture
6 ECTS

Discrete Structures

Instructor Stefan Wolf

Description

This course deals with the mathematics of countable structures. Hereby, central themes are modeling, abstraction, simplification, and generalization. The main topics of the course are propositional logic and proofs; sets, relations, and functions; combinatorics (urn models, inclusion-exclusion), graph theory (trees, planar graphs, Euler tours and Hamilton cycles) and some basic number theory (modular calculus, groups, Euler's theorem, RSA).

Lecture
6 ECTS

Linear Algebra

Instructor Igor Pivkin

Description This course gives an introduction to the field of linear algebra. Concepts and techniques from linear algebra are of fundamental importance in many scientific disciplines and provide the "language" for understanding the behavior of linear mappings and linear spaces. Topics covered are linear systems and Gauss method, vector spaces, linear maps and matrices, determinants, eigenvectors and eigenvalues.

Lecture & Lab
6 ECTS

Programming Fundamentals 2

Instructor Matthias Hauswirth

Description This course teaches how to develop software using an object-oriented approach. It teaches how to structure a problem using the concept of classes, and how to use fields and methods to model state and behavior. The course uses Java as its programming language. It introduces the fundamental concepts of types, dynamic memory allocation, and references. It covers the ideas of collections and iteration to deal with multiple objects, the ideas of inheritance and polymorphism to deal with variability in software, and the idea of exception handling to deal with unexpected situations. It covers principles of design such as coupling and cohesion, encapsulation, and immutability, and it introduces common design patterns. The practical aspects of the course include testing and debugging techniques that help improve the quality of the resulting software.

Prerequisites

- Computer Architecture
- Programming Fundamentals 1

Recommended courses

- Algorithms & Data Structures
- Discrete Structures
- Software Atelier 1: Fundamentals of Informatics
- Software Atelier 2: Human-Computer Interaction

References

- Mandatory textbook: Barnes and Kölling, Objects First with Java (5th edition)

Atélier
6 ECTS

Software Atelier 2: Human-Computer Interaction

Instructor Monica Landoni

Description

This Atelier combines some important ingredients to help students get a better understanding of their future profession as ICT experts. It starts from an introduction to ethics that investigates the many ways and venues ICT can be used maliciously, focuses on responsibilities and proposes ethical solutions. The course moves on to introduce the concept of user centred design to produce usable, useful and used tools. The different stages from ideation to paper prototyping are covered in theory and practice. By putting theory to good use in practice, students will engage in group work to deliver a project to combine their ability to design ethical, usable, useful and enjoyable interfaces. In the final part of the course students will engage in a JAVA lab to test their newly acquired knowledge in GUI design.

References

We will refer to material from seminal books and relevant web sites.

- The UX Book: Process and Guidelines for Ensuring a Quality User Experience; Rex Hartson and Pardha Pyla; Morgan Kaufmann, 2012, ISBN: 0123852412.
- Don't make me think! : a common sense approach to Web usability; Steve Krug; Berkeley, Calif : New Riders Pub.2006.
- The Design of Everyday Things; Donald A. Norman; New York: Basic Books. 2013
- The Evidence-Based User Experience Research, Training, and Consulting web site by the Nielsen Norman group: https://www.nngroup.com

Bachelor of Science

Second year

BSc 2.

Lecture
3 ECTS

Automata & Formal Languages

Instructor Laura Pozzi

Description

The theory of automata and formal languages deals with the problem of modeling computation: what is a computer, and what are its fundamental capabilities? Thus, it constitutes the basis for further studies on the theory of computability and complexity. Additionally, Automata and Formal Languages is a very practical course, as it provides knowledge of the models used in many branches of computer science, from scanners and lexical analyzers in compilers, to programs for designing digital circuits, and even in other areas such as linguistics. At the end of this course you will be familiar with models of computations used today, you will understand how they are fundamental to further studies and you will be ready for a more advanced course on the theory of computation.

References

- Introduction to the Theory of Computation, Michael Sipser, ANY edition

Lecture
6 ECTS

Computer Networking

Instructors Antonio Carzaniga, Silvia Santini

Description

The Internet provides global connectivity for applications and end-users. We want to understand, on the one hand, how common distributed applications such as the world-wide web use the network (the Internet), and on the other hand how the network is designed and how it provides its services to applications. This course serves the designers of distributed applications, as well as network designers as an introduction to advanced studies in computer networking. The course covers the architecture of networked applications and the network itself, their fundamental protocols, and the design principles behind them. This includes applications such as the Web, e-mail, and peer-topeer systems; the two most important transport protocols of the Internet, namely UDP and TCP with its provisions for reliability and congestion control; and the network layer, with the structure of routers and the network as a whole, packet forwarding, and the basics of interdomain and intradomain routing.

Recommended courses

- Programming Fundamentals 2

References

- Textbook: "Computer Networking: A Top-Down Approach", by James F. Kurose and Keith W. Ross, published by Addison-Wesley.

Lecture
6 ECTS

Probability & Statistics

Instructor Davide Eynard

Description

The class provides an introduction to probability theory, descriptive statistics (data exploration and graphical inference; measures of central tendency, dispersion, and correlation; simple and multiple linear regression), and inferential statistics (sampling distributions, confidence intervals, significance testing). Theoretical concepts in the course will be illustrated with real-life examples and datasets, which students will analyse using the R software environment.

References

- William Feller: An introduction to probability theory and its applications. J. Wiley, 1968-1971.
- Steven J. Miller: The Probability Lifesaver. Princeton University Press, 2017.
- Darrell Huff: How to lie with statistics. W.W. Norton & Co, 1954.
- Hogg, Tanis, Zimmerman: Probability and Statistical Inference, Ninth Edition. Pearson, 2015.

Lecture
6 ECTS

Programming Fundamentals 3

Instructor Walter Binder

Description

This course teaches concepts and methods of object-oriented programming as well as concurrent programming techniques. In the first part of the course, the object-oriented programming features of Java are presented in detail, covering inclusion polymorphism, dynamic binding, and parametric polymorphism. The course also teaches design by contract, UML, as well as selected design principles and patterns. The second part of the course gives an introduction to concurrent programming in Java. It covers multi-threading, safety and liveness hazards, and synchronization. The presented techniques enable the development of scalable Java software capable of exploiting modern multi-core hardware.

References

- Design Patterns: Elements of Reusable Object-Oriented Software (1994), by E. Gamma, R. Helm, R. Johnson, J. Vlissides. ISBN-13: 078-5342633610.
- Java Concurrency in Practice (2006), by B. Goetz, T. Peierls, J. Bloch, J. Bowbeer, D. Holmes, D. Lea. ISBN-13: 978-0321349606.

Atélier
9 ECTS

Software Atelier 3: The Web

Instructor Cesare Pautasso

Description

The ultimate goal of the Informatics Atelier is to teach the student to become a computing professional. To this end, the atelier gives an introduction to the role of computing and computer scientists in the professional world as well as society in general and provides an environment for the students to learn about and use specific software tools, work with other students in group projects, and effectively present the results of their projects. The emphasis during the WebAtelier in the third semester is on client/server programming, emerging Web technologies and Web design. The Web Atelier will cover the following Web technologies: REST and HTTP, CSS3, HTML5, JSON and Web Components; students will also learn how to program in JavaScript on the client and on the server-side with Node.JS and the Express framework.

Prerequisites

- Programming Fundamentals 1
- Software Atelier 1: Fundamentals of Informatics

Recommended courses

- Programming Fundamentals 2

References

- Handouts during the theoretical part of the atelier will complement freely available online tutorials.

Lecture
6 ECTS

Data Management

Instructor Patrick Eugster

Description

Databases are essential to applications in a wide variety of domains, including finance, health care, commerce, and telecommunications. In fact, most applications that people use on a day-to-day bases are backed by databases. This course provides a practical introduction to database technology. By the end of this course, students will understand the fundamental concepts about database management systems, become familiar with commercial tools for the design and development of database applications, and be exposed to recent trends in database-like storage systems.
Topics covered include modeling enterprise data with entity-relationship diagrams, the relational model, SQL, logical design with normalization, physical design, query execution, transaction processing, recovery, concurrency, online analytical processing, and NoSQL systems.

References

- Database Management Systems, Ramakrishnan & Gehrke, 3rd ed.
- Fundamentals of Database Systems, Elmasri & Navathe, 6th ed.

Lecture
6 ECTS

Introduction to Computational Science

Instructor Michael Multerer

Description

Numerical computing is an interconnected combination of computer science and mathematics in which we develop and analyze algorithms for solving important problems in science, engineering, medicine, and business---for example, simulating an earthquake , choosing a stock portfolio, or detecting cancer tumors in medical images. The students will learn principles and practices of basic numerical computation. This is a key aspect of scientific computation. This class will cover several topics, including: one-dimensional nonlinear equations; understanding and dealing with sources of error; linear equations and linear least-squares; data fitting; and ordinary differential equations. As much as possible, numerical methods will be presented in the context of real-world applications.

Lecture
6 ECTS

Operating Systems

Instructor Fernando Pedone

Description Operating systems are a fundamental part of any computer system and common to virtually every application. This course surveys conceptual design and implementation issues of such complex programs, starting with the most basic notions of operating systems (e.g., the difference between the kernel and user modes, system calls) and evolving to develop key approaches to operating systems design and implementation. The course delves into the four main pillars of operating systems: process management (i.e., concept of process, multithreaded programming, process scheduling, synchronization, and deadlocks), memory management (i.e., memory-management strategies, virtual memory), storage management (i.e., file systems interface and implementation, mass-storage structure, and I/O systems), and operating systems protection and security. In addition to a conceptual view of operating systems, the course exposes students to the implications of some techniques through a hands-on approach.

References

- Operating System Concepts 9th Edition, A. Silberschatz, P. B. Galvin, and G. Gagne, Wiley, 2012

Lecture
6 ECTS

Systems Programming

Instructor Antonio Carzaniga

Description A "system" integrates functionalities and devices at different levels. Examples are information systems consisting of databases and processing modules, a distributed storage system consisting of networked redundant storage devices, an operating system that manages heterogeneous computing resources, and a robotic system made of physical devices, embedded sensors and controllers, as well as complex processing modules. The most common system programming language is C. This course is a practice-oriented introduction to programming in C and C++. The focus is on features of the language and libraries that are particularly useful in programming systems. This includes the memory model, input/output, the network programming interface and other system calls, the organization of a large system programs, including the relevant language features and the build process, symbols and their relations to compilation units and the linker, and an introduction to symbolic debugging.

Prerequisites

- Programming Fundamentals 2

References

- Textbook (optional): "The C Programming Language", Second Edition. By Brian W. Kernighan and Dennis M. Ritchie. Prentice Hall, Inc., 1988.
- Frequently Asked Questions on C programming. (http://www.c-faq.com/)
- C reference documentation from cppreference.com. (http://en.cppreference.com/w/c)
- C++ reference documentation from cppreference.com. (http://en.cppreference.com/w/cpp)

Atélier
9 ECTS

Software Atelier 4: Software Engineering Project

Instructor

Andrea Mocci, Luca Ponzanelli

Description

Programming skills are essential but not enough to develop large and complex software systems that require the coordination of a team of specialists. Software engineering is about the development of such moderns software systems. The course is about software engineering in practice. Students will learn how to go beyond programming, how to coordinate a team of specialists, how to apply modern methodologies and techniques. Students will experience with state of the art tools to understand the role of automation for developing software systems and coordinate the work of a team.

Prerequisites

- Programming Fundamentals 1
- Programming Fundamentals 2
- Software Atelier 1: Fundamentals of Informatics

Recommended courses

- Algorithms & Data Structures
- Programming Fundamentals 3
- Software Atelier 2: Human-Computer Interaction
- Software Atelier 3: The Web

Bachelor of Science

Third year

Lecture
3 ECTS

Algorithms & Data Structures 2

Instructor Evanthia Papadopoulou

Description This course covers a variety of topics on algorithms and data structures, building upon the material of the first year course “Algorithms and Data Structures”. Algorithms and data structures are fundamental to computer science and the essence of computer programs. The performance of any software system depends on the efficiency of its algorithms and data structures. This course extends the students' knowledge on fundamental algorithms by focusing on several important topics such as data structures for disjoint sets and union-find, interval trees, graphs and graph algorithms such as shortest paths, network flows and matchings, intractability and NP completeness.

References

- Introduction to Algorithms, 3rd edition, by T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, MIT Press, 2009.
- Other useful books (not required): Algorithm Design, by J. Kleinberg, E. Tardos, Addison Wesley, 2005.

Lecture
3 ECTS

Artificial Intelligence

Instructor Luca Maria Gambardella

Description Reasoning, learning, searching for new information, extracting models from knowledge base systems and adapting to unpredictable situations are key factors in any modern computer system. The goal of this course is to investigate knowledge representation models and algorithms that are useful to reason about facts and situations and are suitable to support advanced search and optimisation strategies other than learning systems. In fact, learning from experience and from errors is a crucial aspect for any intelligent system that has to interact with an external environment. The course moves from simple to complex problems introducing concept such as heuristic search and approximation algorithms. These are important tools to allow the student to theoretically analyze and practically solve real life situations.

References

- Artificial Intelligence: a modern approach. S. Russel and Peter Norvig. Prentice Hall,
- Course Material in English will be provided to the students
- Additional readings:
  Artificial Intelligence, third edition, P.H. Winston, Addison-Wesley
- Genetic Algorithms in Search, Optimisation, and Machine Learning, Goldberg, Addison-Wesley, MA

Lecture
6 ECTS

Computer Graphics

Instructors Kai Hormann, Piotr Didyk

Description This course gives a comprehensive introduction to the theoretical and practical aspects of computer graphics. The first half of this course is devoted to the implementation of a ray-tracer, a method for generating pictures of virtual scenes, which is used for special effects and computer-generated movies. A basic version of such a ray-tracer is developed already in the first week. While learning about the theory of local lighting models, colour, homogeneous coordinates, and texture mapping, we keep extending the code until it eventually handles moving objects, shadows, reflections, and refractions. The second half of this course treats the concept of rasterization, an alternative approach to image generation, which is used in games, for example. After implementing our own rasterizer, we learn how to use the OpenGL library and how to program the GPU to achieve special effects. For all programming tasks we provide a framework, so that the students can concentrate on implementing the core methods and algorithm.

References

- Fundamentals of Computer Graphics; Shirley; AK Peters, 2002
- 3D Computer Graphics; Watt; Addison Wesley, 2000
- Computer Graphics with OpenGL; Hearn, Baker; Pearson, 2003
- OpenGL Reference Manual and Programming Guide
- Additional material will be provided through the course homepage.

Lecture
3 ECTS

Experimentation & Evaluation

Instructors Matthias Hauswirth, Marc Langheinrich

Description Computer scientists build complex systems or choose among existing systems to satisfy perceived needs and requirements. The system is then deployed in an environment consisting of humans and other systems. How do we know the impact of the system on the environment and how well it meets the perceived requirements? A fundamental skill in informatics is the ability to design experiments for evaluating computer systems. In this course, the students will acquire a basic understanding of how to design such experiments and what pitfalls to avoid during design and experimentation. Basic concepts of experimental design, data measurement, qualitative and quantitative evaluation, and evaluation with and without users will be covered.

References No mandatory textbook will be used. Handouts will be provided. However, students are encouraged to complement their reading with one or more of the following sources:

- E. J. Davidson: Evaluation Methodology Basics. Sage 2004, 280 pages.
- A. Field, G. Hole: How to Design and Report Experiments. Sage 2003, 384 pages.
- Claes Pohlin et al.: Experimentation in Software Engineering. Springer-Verlag 2012, ISBN-13: 978-3642290435.

Lecture
6 ECTS

Information Retrieval

Instructor Fabio Crestani

Description Today more and more information is becoming available in unstructured or poorly structured form. Examples of information of this type are textual documents, web pages, videos, images, sounds, blogs, etc. The goal of this course is to enable the student to understand the foundations of managing unstructured or poorly structured information. In particular, the aim is to assist students to get an understanding of some of the techniques for the indexing, retrieval, filtering, clustering, and presentation of textual and multimedia information held in digital archives and/or on the web. From this perspective the course complements what the student learned from a previous course on Database technology, where only structured information is managed.

The course will be complemented by practical sessions dealing with the design, implementation, and evaluation of information retrieval systems for medium size collections of documents.

References

- Required: W.B. Croft, D. Metzler, and T. Strohman. Search Engines: Information Retrieval in Practice, Pearson, 2009.
Other books will be suggested during the course, but are not required and could be found in the university library

Lecture
6 ECTS

Numerical Computing

Instructor Olaf Schenk

Description

Numerical computing is an interconnected combination of computer science and mathematics in which we develop and analyze algorithms for solving important problems in science, engineering, medicine, and business---for example, simulating an earthquake , choosing a stock portfolio, or detecting cancer tumors in medical images. The students will learn principles and practices of basic numerical computation. This is a key aspect of scientific computation. This class will cover several topics, including: one-dimensional nonlinear equations; understanding and dealing with sources of error; linear equations and linear least-squares; data fitting; and ordinary differential equations. As much as possible, numerical methods will be presented in the context of real-world applications.

Prerequisites

- Calculus
- Linear Algebra
- Introduction to Computational Science

References

- First Course on Numerical Methods (Computational Science and Engineering), Uri M. Ascher, Chen Greif, SIAM Book, 14. July 2011.
- A Practical Introduction to Programming and Problem Solving Paperback, Stormy Attaway, Matlab, Third Edition, July 1, 2013.

Atélier
9 ECTS

Software Atelier 5: Field Project

Instructors Michele Lanza, Mauro Prevostini

Description

The Field Project Atelier consists of an internship either within a company or within a research group of the faculty. The goal is for the students to obtain hands-on experience with real world problems. The field project atelier can be done individually or as a group, depending on the given context.

List of companies (2018/19):
Best Vision Solutions / Città di Lugano / CodeLounge / Codit / Cryms / Dolphin Engineering / Ex Machina Sagl / Fimax AMS AG / Hegias / Hoxell / Hugo Boss / Lifeware / Mobitrends / O.E. OmniBus Engineering SA

Prerequisites

- Algorithms & Data Structures
- Calculus
- Computer Architecture
- Discrete Structures
- Linear Algebra
- Programming Fundamentals 1
- Programming Fundamentals 2
- Programming Fundamentals 3
- Software Atelier 1: Fundamentals of Informatics
- Software Atelier 2: Human-Computer Interaction
- Software Atelier 3: The Web
- Technical English

Lecture
6 ECTS

Languages & Compilers

Instructor Nate Nystrom

Description Programming languages allow us to express our intentions to computers and to each other. This course teaches you how to analyze programming languages, focusing on semantics, the meaning of programs in languages. To understand the semantics of a programming language, we take an engineering approach, building interpreters and compilers for the language. We use this approach to understand a variety of constructs in functional and objectoriented languages and to understand how these constructs interact with each other in real-world languages.

Prerequisites

- Algorithms & Data Structures
- Automata & Formal Languages
- Computer Architecture
- Programming Fundamentals 1
- Programming Fundamentals 2
- Programming Fundamentals 3
- Systems Programming

References

- "Compilers: Principles, Techniques, and Tools", 2nd edition, Alfred Aho, Monica Lam, Ravi Sethi, and Jeffrey Ullman, 2006.
- "Real World Haskell", Bryan O'Sullivan, Don Stewart, and John Goerzen, 2008.
- "Thinking Functionally with Haskell", Richard Bird, 2015

Lecture
6 ECTS

Machine Learning

Instructor Cesare Alippi

Description Students will learn how to design linear and nonlinear models for regression, prediction and classification as well as assess their performance. At the same time, they will learn how to use deep learning architectures and learning algorithms in key real-world applications. Algorithms for data clustering will be treated as well. Lab sessions will focus on practical aspects and show how to design an appropriate machine learning solution to real-world problems. More in detail, the course will address the following macro topics. Supervised learning: linear and nonlinear models for regression and prediction -also considering recurrent models-, statistical theory of learning, feature extraction and model selection. Deep learning: architectures including autoencoders, convolutional neural networks and learning procedures. Model performance assessment: cross validation, k-fold cross validation, leave-one-out, bootstrap, BLB. Unsupervised learning: K-means clustering, fuzzy C-means, principal component analysis.

Prerequisites

- Calculus
- Linear Algebra
- Probability & Statistics

References

- T.Hastie, R.Tibshirani, J.Friedman, The elements of statistical learning, Springer
- Slides and material provided by the professor

Lecture
6 ECTS

Optimization Methods

Instructors Rolf Krause, Maria Giuseppina Chiara Nestola

Description Optimization is of fundamental importance in virtually all branches of science and technology. As a consequence, optimization methods find their applications in numerous fields, starting from, e.g., network flow and ranging over shape optimization in engineering to optimal control problems. This course provides an introduction into the most important methods and techniques in discrete and continuous optimization. We will present, analyze, implement, and test -along selected problems- methods for discrete and continuous optimization. Particular emphasis will be put on the methodology and the underlying mathematical as well as algorithmic structure. Starting from basic methods as the Simplex method, we will consider different central methods in convex as well as non-convex optimization. This will include optimality conditions, the handling of linear and non-linear constraints, and methods such as interior point methods for convex optimization, Newton's method, Trust-Region methods, and optimal control methods.

References

- Numerical Optimization Authors: Nocedal, Jorge, Wright, S. Springer, 2nd edition, ISBN 978-0-387-40065-5

Lecture
6 ECTS

Theory of Computation

Instructor Natasha Sharygina

Description The class introduces the fundamental mathematical properties of computer hardware, software, and certain applications thereof. It explores what can and cannot be solved on a computer, how quickly, with how much memory, and on which type of computational model. The class is divided into two major parts: computability theory and complexity theory. Computability theory deals primarily with the question of whether a problem is solvable at all on a computer. Complexity theory considers how efficiently the problem can be solved. Two major aspects are considered: time complexity and space complexity, which respectively address a problem of how many steps does it take to perform a computation, and how much memory is required to perform that computation. The subjects have strong connections with engineering practice. Practical exercises will involve experimentation with various tools.

References

- Introduction to the Theory of Computation; Michael Sipser, 2006, second edition (Required)

Project
12 ECTS

Bachelor Project

Instructor Mauro Pezzè

Description

The bachelor project gives the student the opportunity to work independently to develop a solution to a significant (i.e., large) problem. The student learns and demonstrates both independence and a systematic approach to problem solving. The bachelor project gives 12 ECTS, which correspond to the work of 2 typical bachelor-level courses. The students are expected to work throughout the semester under the supervision of their project advisor on the substance of the work, and meet regularly as a group with the Bachelor Project Coordinator to receive instructions about the purpose and mechanics of implementing a long-term project. At the end of the semester, the students produce:

- A project report
- A poster and poster presentation
- A product (if applicable) such as an algorithm, a software library, or application.

Masters in informatics

MSc

4 Semesters' programme
120 ECTS

Informatics

Directors Kai Hormann, Evanthia Papadopoulou

Goals and contents

The Master of Science in Informatics prepares students for current and emerging technologies in computer science by deepening their theoretical knowledge and sharpening their practical skills. The programme is designed for both Bachelor students who wish to complete their education and professionals seeking to refresh their knowledge and sharpen their skills. The Master combines the study of fundamental aspects of computer science with a practical hands-on approach, preparing professionals for successfully pursuing a career in research and development across any application domain.

The Master of Science in Informatics is characterized by a broad offering of topics and subjects that can be freely combined in a learning path tailored to the needs and interests of each student. At USI, students learn how to understand, design, simulate, and optimize complex software-intensive systems. They master the ability to develop automated solutions, introduce them in different business and application domains, and predict and assess their positive impact in the real-world. Students experience the need for a rigorous approach to guarantee the quality of their work while following the most appropriate software engineering methodologies, techniques and state-of-theart tools. Students can benefit from the research excellence of our teaching staff by getting involved in ongoing research activities as part of their master thesis project (which can be carried out across the entire second year of the Master). We offer the unique opportunity to obtain a joint Master's degree in collaboration with University of Milan-Bicocca.

Career opportunities

Informatics is both the infrastructure and the engine of today's society. It plays a key role in industry (pharma, manufacturing of machinery, chemistry, etc.) as well as the service sector (banking, insurance, trade, transport, administration, etc.) in Switzerland. The national training and research institutions have acquired a considerable reputation worldwide, in particular in the field of

Information Technology. Many IT companies, some of them world leaders, have or are planning to have research and development centres in Switzerland. Considering this, graduates in Informatics have excellent opportunities on the job market. The demand for well-educated specialists in Informatics is very high and is expected to grow even more. Graduates of the Master of Science in Informatics are prepared to become, for example, a business-savvy software designer for the highly competitive software industry of the 21st century, a system engineer with the skills to design, build, integrate, validate and maintain reliable, secure, and large distributed systems. Or be trained to solve complex problems in interdisciplinary areas like graphics and special effects, intelligent search engines, computer vision and face recognition, and robotics.

Study plan

The study programme consists of four semesters full-time study (120 ECTS). Students select 24 ECTS of foundational courses (over the two years) and 66 ECTS of electives based on their interests, plus a substantial Master's thesis (30 ECTS).
To broaden the student's perspective, in addition to courses from the other master programmes of the Faculty, up to 6 ECTS of electives can be obtained by following any Master course offered at USI.
A specialisation can be obtained by writing the Master's thesis and taking 18 ECTS of courses in one of the following research areas:

- Computer Systems
- Geometric and Visual Computing
- Information Systems
- Programming Languages
- Theory and Algorithms

	Course	Instructor	ETCS
Fall semester			
Foundational Courses	Algorithms & Complexity	Evanthia Papadopoulou	6
	Distributed Systems	Fernando Pedone	6
	High-Performance Computing	Olaf Schenk	6
	Machine Learning	Jürgen Schmidhuber	6
Electives*	Advanced Programming & Design	Walter Binder	6
	Distributed Algorithms	Fernando Pedone	6
	Mobile Computing	Silvia Santini	6
	Numerical Algorithms	Kai Hormann	6
	Software Engineering	Mauro Pezzè	6
	Software Performance	Matthias Hauswirth	6
	User Experience Design	Monica Landoni, Marc Langheinrich	6
Spring semester			
Foundational Courses	Information Security	Marc Langheinrich	6
Electives*	Advanced Computer Architectures	Laura Pozzi	6
	Advanced Networking	Robert Soulé	6
	Business Process Modeling, Management and Mining	Cesare Pautasso	3
	Compiler Construction	Nate Nystrom	6
	Computational Fabrication	Piotr Didyk	6
	Computer Aided Verification	Natasha Sharygina	6
	Computer Vision & Pattern Recognition	Michael Bronstein	6
	Data Analytics	Fabio Crestani	6
	Geometric Algorithms	Evanthia papadopoulou	6
	Geometric Deep Learning	Michael Bronstein	3
	Quantum Computing	Stefan Wolf	6
	Robotics	Alessandro Giusti	6
Master thesis **		Faculty	30
ETCS Total			120

* Electives from other master programmes of the Faculty of Informatics
** Master Thesis can be started in the 3rd semester.

	Course	Instructor	ETCS
Specialisation in Computer Systems			
Fall	Distributed Algorithms	Fernando Pedone	6
	Mobile Computing	Silvia Santini	6
Spring	Advanced Computer Architectures	Laura Pozzi	6
	Advanced Networking	Robert Soulé	6
	Computer Aided Verification	Natasha Sharygina	6
Specialisation in Geometric and Visual Computing			
Spring	Computational Fabrication	Piotr Didyk	6
	Computer Vision & Pattern Recognition	Michael Bronstein	6
	Geometric Algorithms	Evanthia papadopoulou	6
	Geometric Deep Learning	Michael Bronstein	3
	Robotics	Alessandro Giusti	6
Specialisation in Information Systems			
Fall	Distributed Algorithms	Fernando Pedone	6
	Mobile Computing	Silvia Santini	6
	User Experience Design	Monica Landoni, Marc Langheinrich	6
Spring	Business Process Modeling, Management and Mining	Cesare Pautasso	3
	Compiler Construction	Nate Nystrom	6
	Data Analytics	Fabio Crestani	6
Specialisation in Programming Languages			
Fall	Advanced Programming & Design	Walter Binder	6
	Software Performance	Matthias Hauswirth	6
Spring	Advanced Computer Architectures	Laura Pozzi	6
	Compilers	Nate Nystrom	6
	Computer Aided Verification	Natasha Sharygina	6
Specialisation in Theory and Algorithms			
Fall	Numerical Algorithms	Kai Hormann	6
Spring	Computer Aided Verification	Natasha Sharygina	6
	Geometric Algorithms	Evanthia papadopoulou	6
	Quantum Computing	Stefan Wolf	6

Master of Science
in Artificial Intelligence

4 Semesters' programme
120 ECTS

Artificial Intelligence

Directors Luca Maria Gambardella, Jürgen Schmidhuber

Goals and contents

Artificial Intelligence may not only be the most exciting field in computer science, but of science in general. In fact, the best scientists of the future might even be AIs themselves. Hardware soon will have more raw computational power (CP) than human brains, since CP per cent is still growing by a factor of 100-1000 per decade. And there is no reason to believe that general problem solving software similar to that of humans will be lacking: there already exist mathematically optimal (though not yet practical) universal problem solvers developed at IDSIA. And existing highly practical (but not quite as universal) AI already learn from experience, outperforming humans in more and more fields. For example, biologically plausible deep/recurrent artificial neural networks are learning to solve pattern recognition tasks that seemed infeasible only 10 years ago. Examples: images, handwriting, traffic signs, since 2011 even with superhuman performance - no end in sight. Even creativity has been formalized such that it can now be implemented on machines. The current developments in IS may soon lead to the end of history as we know it (more), and as an IS master student you can become part of this revolution. Artificial Intelligence systems have knowledge, beliefs, preferences and goals, and they have informational as well as motivational attitudes. They observe, learn, communicate, plan, anticipate and commit. They are able to reason about othersystems and their own internal states, to simulate and optimize their performance. AI systems react to dynamic situations adapting their capabilities through learning mechanisms, with a high degree of autonomy.

Career opportunities

Students graduating from this programme will develop a taste for working on complex problems. In their future careers they will be able to apply their knowledge in many interdisciplinary areas including robotics, business forecasting, intelligent search, video games, music and entertainment, chat bots, medical diagnostics, self-driving cars, to name a few.

Study plan

In this master programme a wide variety of techniques will be taught, including intelligent robotics, artificial deep neural networks, machine learning, meta-heuristics optimization techniques, data mining, data analytics, simulation and distributed algorithms. The main courses are integrated with laboratory works where students have the possibility to use real robots and to practice with state of the art tools and methodologies. After the first few lectures of the basic Machine Learning course, AI master students will already know how to train self-learning artificial neural networks to recognize the images and handwritings to the right better than any other known method.

	Course	Instructor	ETCS
First semester			
Core Courses 18 ECTS	Machine Learning	Jürgen Schmidhuber	6
	Deep Learning Lab	Paulo Rauber	3
	Algorithms & Complexity	Evanthia Papadopoulou	6
	Numerical Algorithms	Kai Hormann	3
Electives 12 ECTS	Advanced Topics in Machine Learning	Jürgen Schmidhuber, Alessio Benavoli, Alessandro Giusti	3
	Blockchains - Protocols and Techniques for Distributed Trust	Christian Cachin	3
	High-Performance Computing	Olaf Schenk	6
	Introduction to Ordinary Differential Equations	Michael Multerer	3
	Introduction to Partial Differential Equations	Rolf Krause, Simone Pezzuto	6
	Mobile Computing	Silvia Santini	6
	Programming Styles	Matthias Hauswirth	3
	User Experience Design	Monica Landoni, Marc Langheinrich	6
Second semester			
Core Courses 18 ECTS	Computer Vision & Pattern Recognition	Michael Bronstein	6
	Data Analytics	Fabio Crestani	6
	Stochastic Methods	Illia Horenko	6
	Robotics	Alessandro Giusti	6
Electives 12 ECTS	Advanced Computer Architectures	Laura Pozzi	6
	Advanced Networking	Robert Soulé	6
	Business Intelligence and Applications	Davide Martinenghi	6
	Geometric Algorithms	Evanthia Papadopoulou	6
	Multiscale Methods	Rolf Krause	6
	Philosophy and Artificial Intelligence	Alessandro Facchini	3
	Quantum Computing	Stefan Wolf	6
	Software Atelier: Simulation, Data Science & Supercomputing	Olaf Schenk	6

	Course	Instructor	ETCS
Third semester			
Core Courses 21 ECTS	Artificial Intelligence	Luca Maria Gambardella, Marco Zaffalon	6
	Distributed Algorithms	Fernando Pedone	6
	Master Thesis	Faculty	9
Electives 9 ECTS	Choose from the electives of the 1st semester		
Fourth Semester			
Core Courses 24 ECTS	Computer Vision & Pattern Recognition	Michael Bronstein	6
	Geometric Deep Learning	Michael Bronstein	3
	Master Thesis	Faculty	21
Electives 6 ECTS	Choose from the electives of the 2nd semester		
ETCS Total			120

Master of Science
in Computational Science

4 Semesters' programme
120 ECTS

Computational Science

Director Olaf Schenk

Goals and contents

The Master programme has a unique combination of courses from mathematics and computer science, and additional courses from various applications domains aiming at building deep application-oriented competences in computational science. It has a strong background both in computer science and mathematics and in the development of scientific simulation software. The successful student will acquire strong competences in abstract thinking within a methodology and application oriented education, which will provide the ability to deal with complex models in various applications areas. The students' individual choice of elective courses enables them to tailor the focus of their interdisciplinary personal programme – either method oriented, or computer science-specific. As a result, the programme not only prepares students for current and evolving technologies in computer sciences but will also strongly deepen their knowledge in mathematical and algorithmic methodologies. Along with the mentor, each student will individually set up a study plan for selecting the appropriate elective courses.The Master in Computational Science offers students the opportunity to undertake a double degree programme, in partnership with: INSUBRIA (Università degli Studi dell'Insubria in Como/Varese) or - FAU (Friedrich-Alexander University in Erlangen-Nürnberg) Both dual degree programmes provide dynamic and cross disciplinary training in numerical simulations, applied mathematics, statistics, computer science and data science.

Career opportunities

The multidisciplinary programme offers a streamlined blend of cutting-edge scientific research and practical application, thus providing an excellent foundation for a corporate, industrial, or academic career. Our students receive a firm grounding in programming, mathematical modeling and numerical simulation. The Master in Computational Science opens the doors to industry in software engineering, environmental engineering, financial services, and chemical and pharmaceutical R&D. It is also a strong asset for a PhD in computational science.

Study plan

With the guidance of the Master Director, students will be encouraged to set up an individual study plan that includes appropriate elective courses. The Master Director will advise and accompany students through the entire two-year course of study.

	Course	Instructor	ETCS
First semester			
Mandatory 27 ECTS	High-Performance Computing	Olaf Schenk	6
	Introduction to Ordinary Differential Equations	Rolf Krause, Simone Pezzuto	3
	Introduction to Partial Differential Equations	Michael Multerer	6
	Numerical Algorithms	Kai Hormann	6
	Introduction to Data Science	Ernst Wit	6
Electives 3 ECTS	Software Atelier: Partial Differential Equations	Rolf Krause, Marco Favino	3
	Software Tools in Computational Science	Vittorio Limongelli	3
Second semester			
Mandatory 24 ECTS	Advanced Discretization Methods	Igor Pivkin	6
	Multiscale Methods	Rolf Krause	6
	Software Atelier: Simulation, Data Science & Supercomputing	Olaf Schenk	6
	Stochastic Methods	Illia Horenko	6
Electives 6 ECTS	Advanced Computer Architectures	Laura Pozzi	6
	Bioinformatics	Vittorio Limongelli Daniele Di Marino	6
	Functional and Numerical Analysis (FOMICS)	Rolf Krause	3
	Geometric Algorithms	Evanthia Papadopoulou	6
	Graphical Models and Network Science*	Ernst Wit	6
	Introduction to Network Science	Ernst Wit	6
	USI-CSCS Summer School on Effective High-Performance Computing	Olaf Schenk	6
Third semester			
Mandatory 30 ECTS	Computational Biology & Drug Design	Vittorio Limongelli	6
	Data Assimilation	Sebastian Reich	3
	Generalizations of the Linear Model*	Ernst Wit	3
	Machine Learning	Jürgen Schmidhuber	6
	Molecular Dynamics and Monte Carlo Methods*	Igor Pivkin	6
	Preparation Master's Thesis	Faculty	6
Fourth Semester			
Mandatory 24 ECTS	Master Thesis	Faculty	24
Electives 6 ECTS	Choose from the electives of the 2nd semester		6
ETCS Total			**120**

* This course will not be offered in the academic year 2018/19.

**Master of Science
in Informatics
and Economics**

4 Semesters' programme
120 ECTS

Financial Technology and Computing

Directors Marc Langheinrich, Erik Nowak, Fernando Pedone, Paul Schneider

Goals and contents

The Master of Science in Financial Technology and Computing has been designed to provide graduates with a strong background in informatics with the necessary tools and skills for understanding core challenges in finance while, at the same time, learn about the advanced technology that is needed to drive the next generation finance services.

This Master offers a highly challenging programme that delivers key skills in a novel interdisciplinary domain. A two-tiered structure starts students off with a set of well-balanced core courses from both informatics and finance in the first year, followed by a broad set of electives that can be chosen in the second year, according to personal interests and abilities. A fourth semester Master's thesis that can be done within the context of a university research group, or in collaboration with industry, embedded in our Fintech Laboratory. Since English is the unique teaching language, graduates are well-prepared to work in international companies in Switzerland and beyond.

Career opportunities

The primary labor market for the graduates of the programme is to be found in small Fintech startups, medium to large companies in the finance sector (e.g., banks, insurers, hedge funds) as well as the public sector, both in Switzerland and abroad. Many existing financial companies struggle with keeping up with recent developments in finance technology and thus are in great need of informatics professionals who have a thorough understanding of finance. Potential job profiles range from system architect to system developer to service designer to financial consultant. With a FinTeC master, students will be able to help banks, trading companies, and insurers master this new reality, or, alternatively, be well positioned to challenge existing players with their own startup.

Study plan

This full time programme stretches over two years. It allows students to personalize their study curricula according to their interests. The core skills are acquired in the first two semesters. The third semester is dedicated to more specialized courses and electives that can be chosen according to the students' preference.

	Course	Instructor	ETCS
First semester			
Core Courses 30 ETCS	Financial Econometrics*	Loriano Mancini	6
	Financial Modelling*	Francesco Franzoni	6
	Investments*	Francesco Franzoni	6
	Distributed Systems	Fernando Pedone	6
	High-Performance Computing	Olaf Schenk	6
Second semester			
Core Courses 24 ECTS	Data Analytics	Fabio Crestani	6
	Information Security	Marc Langheinrich	6
	Risk Management*	M.S.E. Garzoli, A. Plazzi	6
	Software Quality & Testing	Mauro Pezzè	6
Electives 6 ECTS	Financial Intermediation	Alberto Plazzi	6
	Derivatives* (required for "Advanced Derivatives")	Giovanni Barone Adesi	6
Third semester			
Core Courses 12 ECTS	Blockchains – Protocols and Techniques for Distributed Trust	Christian Cachin	3
	Artificial Intelligence	Luca Maria Gambardella	6
	FinTech Seminar	Faculty	3
	Master Thesis**	Faculty	6
Electives ** 18/12 ECTS			
Informatics	Distributed Algorithms	Fernando Pedone	6
	Mobile Computing	Silvia Santini	6
	Software Engineering	Mauro Pezzè	6
	Software Performance	Matthias Hauswirth	6
	User Experience Design	Monica Landoni, Marc Langheinrich	6
Finance	Advanced Derivatives*	Giovanni Barone Adesi	3
	Financial Engineering*	Antonio Mele	6
	Fixed Income Markets*	Antonio Mele	6

	Course	Instructor	ETCS
Fourth Semester			
Core Course** 24/30 ECTS	Master Thesis**	Faculty	30
Electives ** 6/0 ECTS			
Informatics	Advanced Computer Architectures	Laura Pozzi	6
	Advanced Networking	Robert Soulé	6
	Business Intelligence and Applications	Davide Martinenghi	6
	Business Process Modeling, Management and Mining	Cesare Pautasso	3
	Compiler Construction	Nate Nystrom	6
	Computer Aided Verification	Natasha Sharygina	6
	Software Architecture	Cesare Pautasso	6
	Software Atelier: Simulation, Data Science & Supercomputing	Olaf Schenk	6
Finance	Critical Consumer Behaviour	Michael Gibbert	6
ETCS Total			120

* To obtain the SFI accreditation, students have to achieve 45 ECTS among these courses.

** Students choose 18 ECTS of electives from the Informatics and Finance courses listed, and from other courses from the Master programmes offered by the Faculty of Informatics and the Faculty of Economics (upon approval of the Master's director).

Master of Science
in Informatics
and Economics

4 Semesters' programme
120 ECTS

Management and Informatics

Directors

Marc Langheinrich, Nikolaus Beck

Goals and contents

This Master offers a balanced combination of courses that cover the necessary back-ground in management, fundamental aspects of current and evolving IT, as well as specialised topics at the interface between management and informatics, such as enterprise resource planning. Since the programme is taught entirely English, graduates are well prepared to work in international companies. Moreover, the interdisciplinary approach of this Master provides a general skill to work across traditional areas. This full-time programme stretches over two years. It allows students to personalise their study curricula according to their interests. The first year focuses on the acquisition of foundational knowledge. Students who obtained a Bachelor's degree in informatics or a related field (mathematics, engineering, physics, etc.) enter the programme in the Informatics track and follow a set of courses that provide them with a solid background in management disciplines. In contrast, the Management track targets students with a background in economics or management, and teaches the fundamental principles of informatics. In addition, all students attend mandatory courses that cover the interface between management and informatics. The second year offers specialised courses and electives that students can choose according to their preferences. A mandatory practical field project lets student gain practical consulting experience by working for real clients in small project teams. A substantial master's thesis concludes the programme.

Career opportunities

On the one hand, graduates from this Master will have sufficient knowledge in informatics to be able to interact with the IT department of an organization. A profound understanding of the technical issues involved gives graduates the ability to both evaluate technical proposals and articulate possible solutions to the organization or the customer. On the other hand, graduates will also understand the tactical and strategic use of IT to enhance the efficiency of an organization, or how to explain user requirements in terms that can be understood by the IT department or the client. Graduates of the programme will find work in medium to large companies, as well as the public sector, both in Switzerland and abroad. Most companies struggle with integrating IT in their organization, so people who can be the interface between the technical and organizational parts of a company are in great demand. Potential job profiles range from project management to consulting and include areas such as: evaluating the benefits, and managing the introduction, of a new technology into an organization; designing and implementing small and large scale IT systems; and consulting companies and customers regarding requirements and limitations of particular IT systems.

Study plan

This full time programme stretches over two years. It allows students to personalize their study curricula according to their interests. The basic knowledge is acquired in the first two semesters. Students who obtained a Bachelor's degree in informatics or a related field (mathematics, engineering, physics, etc.) enter the programme in the Informatics track and follow a set of courses that provide them with a fundamental insight into the management disciplines. In contrast, the Management track is tailored for students with a background in economics or management and teaches the basic aspects of informatics. In addition, all students attend mandatory courses that cover the interface between management and informatics. The third and fourth semester are dedicated to specialized courses and electives that can be chosen according to the students' preference. Moreover, the students participate in a practical field project, which is done in groups for a real company, and conclude their studies by writing a substantial master's thesis.

	Course		Instructor	ETCS
First semester				
Core Courses	Enterprise Resource Planning		Daniel Florian	6
12 ETCS	Enterprise Resource Planning Lab		Cinzia Cappiello	3
	Project Management		Paulo Gonçalves	3
Informatics	Accounting		Stefano Calciolari	6
track	Corporate Strategy		Matteo Prato	6
18 ETCS	Orthodox and Critical Perspectives in Marketing		Roberta De Sanctis, Monica Mendini, Luca Visconti	6
Management	Fundamentals of Informatics		Natasha Sharygina	6
track	Introduction to Programming		Walter Binder	6
18 ETCS	Probability & Statistics		Davide Eynard	6
Second semester				
Core Courses	Business Intelligence and Application		Davide Martinenghi	6
12 ETCS	Business Process Modeling, Management and Mining		Cesare Pautasso	3
	Operations Management		Paulo Gonçalves	3
Informatics	Decision Making		Dirk Martignoni	3
track	Entrepreneurship: Theory and Practice		Gianluca Colombo	3
6 ETCS				
Management	Databases		Fabio Crestani	18
track				
18 ETCS				
Electives	Critical Consumer Behaviour	COM	Michael Gibbert	6
12 ETCS	Human Resources Management	ECO	Luca Solari	3
	Innovation	ECO	Natasha Vijay Munshi	3
	International Business	ECO	Francesco Ciabuschi	3
	Mergers and Acquisition	ECO	E.L.M. Bettinazzi	3
	Organizational Learning	ECO	Nikolaus Beck	6
	Writing Business Plans*	ECO	Gianluca Colombo	3
	Information Security	INF	Marc Langheinrich	6
	Robotics	INF	Alessandro Giusti	6
	Software Architecture	INF	Cesare Pautasso	6
	Software Quality & Testing	INF	Mauro Pezzè	6

	Course		Instructor	ETCS
Third semester				
Core Courses 6 ETCS	Six Sigma		Paolo Rossetti	6
Capstone Work 12 ETCS	Field Project		Marc Langheinrich	12
Electives 12 ECTS	Analytical Thinking	ECO	Léna Pellandini-Simány	3
	Business Dynamics	ECO	Paulo Gonçalves	3
	Digital Challenges in Marketing and Big Data	ECO	Andreina Mandelli	3
	Service Design Marketing	ECO	Andreina Mandelli	3
	Organizational Design & Change	ECO	Francesca Pallotti	3
	Distributed Systems	INF	Fernando Pedone	6
	Machine Learning	INF	Jürgen Schmidhuber	6
	User Experience Design	INF	Monica Landoni, Marc Langheinrich	6
Fourth Semester				
Capstone Work 18 ETCS	Master Thesis		Faculty	18
Electives 12 ECTS	Choose from the electives of the 2nd semester			12
ETCS Total				120

* This course will not be offered in the academic year 2018/19.

Master of Science
in Software & Data Engineering

4	Semesters' programme
120	ECTS

Software & Data Engineering

Directors Cesare Pautasso, Gabriele Bavota

Goals and contents

Software plays a pivotal role in almost all aspects of our life, including transportation, communication, economy, and healthcare. We put trust in software to accomplish complex and vital tasks for us, such as managing our finances, sharing our family and friends' memories, diagnosing diseases, flying airplanes or driving cars. The complexity of these tasks, while becoming transparent to us, does not go away: it is distilled into the software our civilization depends on. Indeed, we are already in the era of ultra-large-scale software systems, composed by millions of code components interacting among them. In such a scenario, software cannot be understood without its data and data becomes valuable only thanks to the software analyzing it. In other words, software engineering aims at managing the complexity of software, keeping it under control. Data engineering focuses instead on how to collect, store, and process huge amounts of data, that can be analyzed to gather insights and support decision making activities. The master features courses taught by world's leading researchers of the Software Institute at the USI Faculty of Informatics.

Career opportunities

Data is the new natural resource to be mined and exploited using software. Data analytics software provides actionable insights at the basis of continuous improvement and decision making processes. Such insights can be found by exploring large quantities of data, by asking the right questions and knowing how to reliably and efficiently find the appropriate answers. Students graduating in this Master will be highly specialized software and data engineers, able to fully understand and manage the complexity of modern software systems and of the sea of data surrounding them. Mastering how to effectively use software to deal with the data deluge is a key capability for any organization undergoing digital transformation efforts. Also, the demand for software and data engineers is currently very high and it is expected to grow even more in the near future. Besides the expected high employability in industry, graduated students will also represent the perfect candidate for pursuing a PhD degree at USI, by working in one of the research groups of the Software Institute.

Study plan

The study programme is compounded of four modules: Software Engineering, Data Engineering, Electives, and Master thesis.
The Software Engineering module embraces 36 ECTS and provides students with a deep knowledge of state-of-the-art techniques. Topics related to this module are software design, software quality and testing, software architecture, software performance, and software analytics.
The Data Engineering module includes three courses (18 ECTS) teaching students techniques and tools to design and model data (1st semester), to convert data into information (2nd semester), and to transform information into knowledge useful to support decision making activities (3rd semester). The topics studied in the Software and the Data Engineering modules are continuously integrated through the whole course of study. This is done by devoting 18 ECTS to deal with both Software and Data Engineering with project based learning.
The Electives module includes 12 ECTS, that the student can freely select from a given list of courses offered at the USI Faculty of Informatics based on his/her personal preference. Finally, the remaining 36 ECTS are dedicated to the MSc thesis.
Students will use the 6 ECTS of the 3rd semester to visit the research groups of the Software Institute of the Faculty of Informatics and to prepare a thesis proposal. Then, they will work full time on the thesis in the 4th semester in the research group of their choice.

	Course	Instructor	ETCS
First semester			
Mandatory 24 ECTS	Software Design & Modeling	Carlo A. Furia	6
	Engineering of Domain Specific Languages	Andrea Mocci	3
	Programming Styles	Matthias Hauswirth	3
	S&DE Atelier: Design 101	Michele Lanza	6
	Data Design & Modeling	Marco Brambilla	6
Electives 6 ECTS	Software Engineering	Mauro Pezzè	6
	Mobile Computing	Silvia Santini	6
Second semester			
Mandatory 24 ECTS	Software Analysis	Carlo A. Furia	6
	Software Architecture	Cesare Pautasso	6
	S&DE Atelier: Visual Analytics	Marco D'Ambros	6
	Information Modeling & Analysis	Paolo Tonella	6
Electives 6 ECTS	Advanced Networking	Robert Soulé	6
	Compiler Construction	Nate Nystrom	6
	Software Quality & Testing	Mauro Pezzè	6
Third semester			
Mandatory 30 ECTS	Software Analytics	Gabriele Bavota	6
	Software Performance	Matthias Hauswirth	6
	Knowledge Analysis & Management	Marco Brambilla	6
	S&DE Atelier: Internet of Things	Luca Mottola	6
	Software & Data Engineering Seminar	Faculty	6
Fourth Semester			
Mandatory 30 ECTS	Master Thesis	Faculty	30
ETCS Total			**120**

PhD programme

PhD programme

The PhD programme of the Faculty of Informatics at the Università della Svizzera italiana promotes the development of professionals interested in academic or industrial research. A successful PhD student will gain a broad knowledge and understanding of the general field of informatics, as well as an in-depth specialisation in an area of interest. Working with one or more members of the Faculty, who serve as the student's advisors, the student will learn the methods and practical skills to conduct research, and will contribute original, useful, and scientifically valid ideas in their chosen area of interest. PhD students are also encouraged to explore other areas and to interact and collaborate with other students and professors within the Faculty as well as in the broader research community. At present the Faculty awards the following qualifications: PhD in Informatics and PhD in Computational Science. Most students in the PhD programme are supported as assistants. The support covers tuition and provides a stipend. Responsibilities of assistants include both teaching and research duties. Generally students receive support as long as funds are available and the student is making adequate progress through the programme (as described in the regulations). The PhD programme is governed by regulations adopted by the Faculty: www.inf.usi.ch/regolamenti_tutti.htm In order to be admitted, the applicant must have completed a Masters degree in computer science, informatics, or a closely related field prior to joining the programme (but not necessarily prior to applying to the programme). For more information regarding the admission to the programme: www.inf.usi.ch/dottorato_regolamenti.htm

Study plan
The Faculty of Informatics offers PhD courses to students pursuing a PhD at the Faculty. The course Introduction to Doctoral Studies is mandatory for first year PhD students. Master courses may be cross- listed as PhD courses

Lecturers’ profiles

Full professors

Antonio Carzaniga
Antonio Carzaniga joined the Faculty of Informatics at USI when the Faculty was founded in 2004. From 2001 to 2007 he was also an assistant research professor in the Department of Computer Science at the University of Colorado at Boulder, USA. Antonio received the Laurea degree in electronics engineering and the Ph.D. degree in computer science from Politecnico di Milano, Italy. Antonio is a curious researcher. His primary research interests are in the areas of distributed systems and software engineering, specifically in content-based networking, information-centric networking, distributed publish/subscribe systems, middleware, software adaptability and automatic fault tolerance, and testing. He also conducted research in software configuration management and code mobility. Antonio is also a dedicated and passionate teacher. He has developed and taught a number of courses in the Faculty of Informatics at USI, including Algorithms and Data Structures, Computer Networking, and Systems Programming.
antonio.carzaniga@usi.ch

Cesare Alippi
Cesare Alippi was awarded the degree in electronic engineering cum laude and PhD from Politecnico di Milano, Italy.
IEEE Fellow; Board of Governors member, International Neural Network Society; Board of Directors member, European Neural Network Society; Past Vice-President, IEEE Computational Intelligence Society; AE IEEE Computational Intelligence Magazine, IEEE-Trans. Instrumentation and Measurements, IEEE-Trans. In 2016 he received the International Neural Networks Society Gabor Award and the Outstanding IEEE Transactions on Neural Networks and Learning Systems Paper Award; the 2013 IBM Faculty award; the 2004 IEEE Instrumentation and Measurement Society Young Engineer Award. Current research activity addresses adaptation and learning in non-stationary environments and Intelligence for embedded, cyber-physical systems and IoT. He holds 5 patents, has published one monograph book, 6 edited books and about 200 papers in international journals and conference proceedings.
alippi@elet.polimi.it

Walter Binder
Walter Binder is a professor in the Faculty of Informatics, Università della Svizzera italiana (USI), Switzerland. He holds an MSc, a PhD, and a Venia Docendi from TU Wien, Austria. Before joining USI, he was a post-doctoral researcher in the Artificial Intelligence Laboratory, École Polytechnique Fédérale de Lausanne, Switzerland. His main interests are in the areas of program analysis, virtual machines, parallel programming, and Cloud computing.
walter.binder@usi.ch

Michael Bronstein
Michael Bronstein received the Ph.D. in computer science (2007) from the Technion in Israel. His main research interests are geometric methods in computer vision, pattern recognition, and computer graphics. Prof. Bronstein's research was featured in international news and recognized by several awards, including three ERC grants, Google faculty award, Radcliffe fellowship from Harvard University and Rudolf Diesel fellowship from TU Munich. He has served on program committees of major conferences in his field and was keynote speaker in numerous international symposia. Prof. Bronstein is also actively involved in technology transfer and consulting. His start-up track record includes Novafora (2004-2009 as co-founder and VP of video technology) and Invision (2009-2012 as one of principle technologists). Since the acquisition of Invision by Intel in 2012, Michael has also served as a Research Scientist and Principal Engineer at Intel, where he was one of the key algorithm developers for the RealSense 3D sensor.
michael.bronstein@usi.ch

Fabio Crestani
Fabio Crestani is a Full Professor at USI since 2007. Previously he was Professor (2000-06) at the University of Strathclyde (UK) and Assistant Professor (1992-97) at the University of Padua (Italy). In between he was Research Fellow at the University of Glasgow (UK), at the International Computer Science Institute in Berkeley (USA), and at the Rutherford Appleton Laboratory (UK). Recently he received a Chair of Excellence at the University Carlos III in Madrid (2011-12), a Visiting Scholarship at Yahoo! Labs (2014), and a Visiting Professorship at the UPMF in Grenoble (2015). Fabio holds a degree in Statistics and Economics from the University of Padua (Italy) and a MSc and PhD in Computing Science from the University of Glasgow (UK). He leads the local Information Retrieval and Text Mining group (see http://www.ir.inf.usi.ch/ for details).
fabio.crestani@usi.ch

Patrick Eugster
Patrick Eugster joined USI as a Full Professor Computer Science in 2017. He is also an Adjunct Faculty at Purdue University (where he was a regular faculty member 2005-2016) and at TU Darmstadt (2014-2017). Patrick holds M.S. (1998) and Ph.D. (2001) degrees from EPFL. Patrick is interested in software systems, with a particular focus on distributed systems and programming models/languages, and the intersection between the two. He has co-authored over 120 scientific articles on these topics. His research has been awarded by various funding agencies (e.g., US NSF CAREER 2007, DARPA Computer Science Study Group 2011, ERC Consolidator 2014) and companies (e.g., Google Research Award 2003, NetApp Faculty Fellowship 2014).
patrick.eugster@usi.ch

Ilia Horenko
Illia Horenko is full professor in the faculty of informatics and the Institute of Computational Science of the Università della Svizzera italiana. He received a Ph.D. in applied mathematics from the Free University (FU) of Berlin in 2004 and spent several years as a postdoctoral research fellow at the Biocomputing Group and Climate Research Group at the FU Berlin, before joining the faculty of mathematics and computer science of the FU Berlin as an assistant professor in 2008.
His research interests are focussed on the development and practical implementation of data analysis algorithms and time series analysis approaches. Published applications of the methods developed by I. Horenko include problems from climate research, economics, biophysics, engineering and bioinformatcs. Prof. Horenko has published over 50 papers in the professional literature. He was a co-organizer of several big scientific programs and is a frequent reviewer for international funding agencies and the top journals in his field.
illia.horenko@usi.ch

Kai Hormann
Kai Hormann is a full professor in the Faculty of Informatics at USI. He received a PhD in computer science from the University of Erlangen in 2002 and spent two years as a postdoctoral research fellow at Caltech in Pasadena and the CNR in Pisa, before joining Clausthal University of Technology as an assistant professor in 2004. During the winter term 2007/2008 he visited Freie Universität Berlin as a BMS substitute professor and came to Lugano as an associate professor in 2009. His research interests are focussed on the mathematical foundations of geometry processing algorithms and their applications in computer graphics and related fields. In particular, he is working on generalized barycentric coordinates, subdivision of curves and surfaces, barycentric rational interpolation, and dynamic geometry processing. Professor Hormann has published over 70 peer-reviewed papers and is an associate editor of Computer Aided Geometric Design, Computers & Graphics, and the Dolomites Research Notes on Approximation.
kai.hormann@usi.ch

Rolf Krause
Rolf Krause is chair of advanced scientific computing and the director of the institute of computational science in the faculty of informatics. From 2003 to 2009, he was professor at the University of Bonn. During that time he spent a sabbatical at UC San Diego (USA) and Columbia University New York (USA). In 2002 he was on a Postdoctoral research visit at the Courant Institute (NYU, New York). He holds a Diploma and a PhD (2000) in Mathematics from the FU Berlin (Germany). His research focuses on numerical simulation and mathematical modeling in the life sciences, in particular medicine, and for engineering applications. A focal point of his work is the development and implementation of parallel simulation-methods, which show excellent performance also in real world applications. He is member of the editorial board of the SIAM Journal on Scientific Computing (SISC) and of the Springer Journal Computing and Visualization In Sciene (CVS).
rolf.krause@usi.ch

Marc Langheinrich
Prof. Langheinrich received his PhD (Dr. sc. ETH) on the topic of "Privacy in Ubiquitous Computing" from the ETH Zurich, Switzerland, in 2005. He has published extensively on both privacy and usability of ubiquitous and pervasive computing systems, and is a regular program committee member of various conferences and workshops in the areas of pervasive computing, security and privacy, and usability. Marc is an Associate Editor in Chief of IEEE Pervasive Computing Magazine, and a Steering Committee member of the UbiComp and IoT conference series. In 1997, Marc spent a year as a Fulbright Scholar at the University of Washington, Seattle, where he also completed his thesis work in the fields of information retrieval and software agents. From 1997 to 1999 he lived in Tokyo, Japan, where he worked at NEC Research on projects involving personalization and electronic commerce. Prof. Langheinrich joined USI in 2008.
marc.langheinrich@usi.ch

Michele Lanza
Michele Lanza is full professor at the faculty of informatics of the Università della Svizzera italiana, Switzerland, which he co-founded in 2004. His doctoral dissertation, completed in 2003 at the University of Bern, received the Denert award for best thesis in software engineering of 2003. Prof. Lanza received the Credit Suisse Award for best teaching in 2007 and 2009. At the Università della Svizzera italiana Prof. Lanza directs the Software Institute (http://si.usi.ch) and leads the REVEAL research group, working in the areas of software visualization, evolution, and analytics. He authored more than enough articles and the book "Object-Oriented Metrics in Practice". Prof. Lanza is involved in a number of scientific communities, and has served on close to a 100 program committees. He likes (software) visualization, universal design, and when it comes to programming he thinks it's nothing but Smalltalk.
michele.lanza@usi.ch

Evanthia Papadopoulou
Evanthia Papadopoulou is a professor of computer science at the Università della Svizzera italiana. From 1996 to 2008 she had been a research staff member at the IBM T.J. Watson research center, Yorktown Heights NY, USA. She had also been a Faculty member in the department of computer science at the Athens University of Economics and Business. She holds a BS degree in mathematics from University of Athens, an MS in computer science from University of Illinois at Chicago, and a Ph.D. in computer science from Northwestern University, USA, December 1995. Her research interests include the design and analysis of discrete algorithms, computational geometry and its applications, robust geometric computing and the implementation of geometric algorithms, algorithmic aspects of VLSI computer-aided design. For her work on "Voronoi diagram based VLSI Critical Area Analysis", she received the IBM outstanding innovation award, August 2006, and the Technical Accomplishment for IBM Research, December 2006
evanthia.papadopoulou@usi.ch

Michele Parrinello
Professor Parrinello is known for his many technical innovations in the field ofatomistic simulations and for a wealth of interdisciplinary applications rangingfrom materials science to chemistry and biology. Together with Roberto Car heintroduced ab-initio molecular dynamics, also known as the Car-Parrinello method, marking the beginning of a new era both in the area of electronic structure calculations and in molecular dynamics simulations. He is also known for the Parrinello-Rahman method, which allows crystalline phase transitionsto be studied by molecular dynamics. More recently he has introduced metadynamics for the study of rare events and the calculation of free energies. Forhis work he has been awarded many prizes and honorary degrees. He is amember of numerous academies and learned societies, including the German Berlin-Brandenburgische Akademie der Wissenschaften, the British Royal Societyand the Italian Accademia Nazionale dei Lincei, which is the major academyin his home country Italy. Born in Messina in 1945, he got his degree from the University of Bologna and is currently professor of Computational Sciencesat Università della Svizzera italiana (Faculty of Informatics) and ETHZ in Switzerland.
michele.parrinello@usi.ch

Cesare Pautasso
Cesare Pautasso is full professor at the USI Faculty of Informatics. Previously he was a researcher at the IBM Zurich Research Lab and a senior researcher at ETH Zurich, where he earned his PhD in 2004. His research group is interested in interdisciplinary research around all aspects of the architecture, design and engineering of next-generation Web information systems, focusing on building experimental systems to explore the intersection of model-driven software composition techniques, RESTful business process management, liquid software architectures, and edge computing. He edits the IEEE Software Insights department. He was the general chair of the 16th International Conference on Web Engineering (ICWE2016) held in Lugano in June 2016. He regularly referees for Swiss, EU and International funding agencies. Since 2010 he is an advisory board member of Enterprise Web. You can follow him on http://design.inf.usi.ch
@pautasso and
@pautasso@scholar.social
cesare.pautasso@usi.ch

Fernando Pedone
Fernando Pedone is a full professor in the Faculty of Informatics at the Università della Svizzera italiana (USI), Switzerland, and has been also affiliated with Cornell University (as a visiting professor), EPFL, and Hewlett-Packard Laboratories (HP Labs). He received the Ph.D. degree from EPFL in 1999. His research interests include the theory and practice of distributed systems and distributed data management systems. Fernando Pedone has authored more than 100 scientific papers and 6 patents. He is co-editor of the book "Replication: theory and practice", Springer 2010.
fernando.pedone@usi.ch

Mauro Pezzè
Mauro Pezzè is a professor of software engineering at the Faculty of Informatics of USI. Mauro Pezzè has served as associate editor of IEEE Transactions on Software Engineering from 2013 to 2018 and has served as associate editor of ACM Transactions on Software Engineering and Methodology from 2006 to 2012. He has been program co-chair of the International Conference on Software Engineering (ICSE 2012), general chair (2013) and program chair (2006) of the ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA), program chair of the International Conference on Fundamental Approaches to Software Engineering (ETAPS-FASE 2003). Pezzè's research interests are in software engineering and in particular in the areas of software testing and analysis and self-healing and autonomic systems. He is the co-author of more than 160 papers appeared in the top journals and in the proceedings of the main conferences in software engineering and of a book on Software Testing and Analysis.
mauro.pezze@usi.ch

Laura Pozzi
Laura Pozzi joined the Faculty of Informatics at USI as Assistant Professor in 2005, and she is now a Full Professor. Before landing at USI, Laura was a postdoctoral researcher in the Processor Architecture Laboratory of EPFL, a research engineer with STMicroelectronics in the Silicon Valley in California, and an Industrial Visitor at UC Berkeley. Laura holds a Master (1996) and a Ph.D. degree (2000) in Computer Engineering from Politecnico di Milano, Italy. Her research interests include embedded processor customization, compiler techniques for customizable processors, coarse-grain reconfigurable fabrics, and in general application-mapping onto unconventional architectures. Laura currently teaches two courses at USI: Automata and Formal Languages in the Bachelor; Advanced Computer Architectures in the Master.
laura.pozzi@usi.ch

Olaf Schenk
Olaf Schenk is a full professor at the Institute of Computational Science within the Faculty of Informatics at the Universita della Svizzera italiana, Switzerland. He graduated in Applied Mathematics from Karlsruhe Institute of Technology (KIT), Germany, and earned his PhD from the Department of Information Technology and Electrical Engineering of ETH Zurich. He conducts research in applied algorithms, computational science, and software tools for high-performance scientific computing. He is a recipient of an IBM faculty award (2008) and two leadership computing awards from the U.S. Department of Energy (2012, 2013). He serves on the editorial board of the SIAM Journal for Scientific Computing and has been elected as the Program Director for the SIAM Activity group on Supercomputing (2016-2017) and as the current Vice Chair for the SIAM Activity group on Supercomputing (2018-2019). He regularly is a chair of international computer science conferences (e.g. ACM PASC, SIAM PP, IEEE CSE).
olaf.schenk@usi.ch

Jürgen Schmidhuber
Since age 15 or so, Prof. Jürgen Schmidhuber's main scientific ambition has been to build an optimal scientist through self-improving Artificial Intelligence, then retire. His AI team has won nine international competitions in machine learning and pattern recognition (more than any other AI research group worldwide) and six independent best paper/best video awards, achieved the world's first superhuman visual classification results, and established the field of mathematically rigorous universal AI and optimal universal problem solvers. His formal theory of creativity & curiosity & fun explains art, science, music, and humor. He generalized algorithmic information theory and the many- worlds theory of physics. Many famous leading companies are now using the machine learning techniques developed in his group at the Swiss AI Lab IDSIA (a Business Week Top 10 AI Lab) & USI & SUPSI. He published 333 peer-reviewed papers, and is recipient of the 2013 Helmholtz Award of the International Neural Networks Society.
juergen@idsia.ch

Natasha Sharygina
Prof. Sharygina's research focuses on improving the program development process through formal methods of specification and verification. In particular, her interests lie in automated formal verification with a specific focus on software/hardware model checking, information security, static analysis, abstract interpretation and decision procedures. Prof. Sharygina received a Ph.D. degree from the University of Texas at Austin, USA in 2002. Her professional experience includes consulting at Bell Labs, Lucent Technologies Computing Sciences Research in 2000-2001 and a research faculty position at Carnegie Mellon University, SEI in 2002-2005. Prof. Sharygina directs the USI Formal Verification and Security group (www.verify.inf.usi.ch) whose work resulted in award-winning theoretical frameworks and practical tools to enable sound and scalable verification of industrial-size systems.
natasha.sharygina@usi.ch

Paolo Tonella
Paolo Tonella is Full Professor at the Faculty of Informatics and at the Software Institute of Università della Svizzera Italiana (USI) in Lugano, Switzerland. He is also Honorary Professor at University College London, UK. Until mid 2018 he has been Head of Software Engineering at Fondazione Bruno Kessler, Trento, Italy. Paolo Tonella holds an ERC Advanced grant as Principal Investigator of the project PRECRIME. Paolo Tonella wrote over 150 peer reviewed conference papers and over 50 journal papers. His H-index (according to Google scholar) is 52. He is/was in the editorial board of the ACM Transactions on Software Engineering and Methodology, of the IEEE Transactions on Software Engineering, of Empirical Software Engineering, Springer, and of the Journal of Software: Evolution and Process, Wiley. Paolo Tonella teaches Information Modeling and Analysis at the Master in Data and Software Engineering.
paolo.tonella@usi.ch

Ernst Wit
Ernst Wit joined the Faculty of Informatics at USI as full professor in June 2018. He obtained PhDs in Philosophy (1997) and Statistics (2000) in the US. From 2000 until 2005 he was in the Statistics Department at the University of Glasgow, where he became a Reader. In 2005 he became head of the Medical Statistics Unit (12 FTE) at the University of Lancaster as full professor and as Director implemented a thriving Master in Statistics programme. From 2008 to 2018 Wit was at the University of Groningen, where he continued to work on methodological development in high-dimensional inference with a specific focus on network modelling. He is the author of 105 peer-reviewed publications, and has served as the President of the European Bernoulli Society and as member of the Board of Directors of the International Biometrics Society. He advises the Ministry of Internal Affairs in the Netherlands on statistical matters relating to elections and referendums since 2014.
ernst.wit@usi.ch

Stefan Wolf
Stefan Wolf joined USI's Informatics Faculty in 2011 as an Associate Professor, and is Full Professor since 2015. Born in Schaffhausen, Switzerland, he received a Dipl. Math. ETH, followed by a Ph.D. in Computer Science in the field of Informatiion-Theoretic Cryptography from ETH Zurich under the supervision of Professor Ueli Maurer. After a postdoc at McGill University, Montreal, he was Assistant Professor at University of Waterloo, Ontario, and Université de Montréal, Quebec. From 2005 to 2011, he was a Swiss National Science Foundation (SNF) Professor for Quantum Information at the Computer Science Department of ETH Zurich. His research domain lies in the fields of cryptography, information theory, and quantum information processing.
stefan.wolf@usi.ch

Associate professors

Carlo A. Furia
Carlo A. Furia is an associate professor at the Software Institute in the Faculty of Informatics of the Università della Svizzera Italiana (USI). His research interests center around developing rigorous yet practical techniques and tools to help improve the quality, correctness, and reliability of software and systems; much of his work aims at making verification and formal methods practical and widely applicable. He publishes in software engineering, formal methods, and verification venues. He has a PhD in computer science from the Politecnico di Milano; Before joining USI, he was an associate professor at Chalmers University of Technology. Before Chalmers, he spent several years as senior researcher at ETH Zurich.
carlo.alberto.furia@usi.ch

Matthias Hauswirth
After an apprenticeship as an "Elektroniker" in Thun, Matthias Hauswirth received an "Informatik Ing. HTL" from Ingenieurschule Biel and later a Masters and a PhD from the University of Colorado at Boulder. During his studies he spent several summers as an intern at Microsoft Research and at the IBM T. J. Watson Research Center. Right after his graduation, in Fall 2005, he joined the Faculty of Informatics here at USI. In 2015/2016 he spent a sabbatical at SRI International to learn more about learning. Matthias is interested in the area between programming languages, runtime systems, and software engineering, with a focus on the efficiency of software, its users, and its developers. He loves to help students learn, and he is interested in finding approaches to increase the efficiency and the effectiveness of teaching informatics.
matthias.hauswirth@usi.ch

Igor Pivkin
Igor Pivkin received his B.Sc. and M.Sc. degrees in Mathematics from Novosibirsk State University, Russia, M.Sc. degree in Computer Science and Ph.D. in Applied Mathematics from Brown University, USA. Before coming to Lugano, he was a Postdoctoral Associate in the Department of Materials Science and Engineering at Massachusetts Institute of Technology, USA. The research interests of Igor Pivkin lie in the area of multiscale/multiphysics modeling, corresponding numerical methods and parallel large-scale simulations of biological and physical systems. Specific areas include biophysics, cellular and molecular biomechanics, stochastic multiscale modeling, and coarse-grained molecular simulations.
igor.pivkin@usi.ch

Silvia Santini
Silvia Santini is since September 2016 an Associate Professor at the Faculty of Informatics of USI.
From July 2014 until August 2016, Silvia held an Associate Professor position at TU Dresden, where she led the Embedded Systems Lab.
From October 2011 until July 2014

she was an Assistant Professor at the Department of Electrical Engineering and Information Technology of TU Darmstadt, Germany, where she led the Wireless Sensor Networks Lab. Previously, from 2009 until 2011, she was a postdoctoral researcher in Prof. Friedemann Mattern's Distributed Systems Group at the Department of Computer Science of ETH Zurich, Switzerland. From November 2010 until February 2011 she joined Leonidas Guibas's research group at Stanford University as a visiting scholar. Silvia completed her PhD Thesis under the supervision of Prof. Friedemann Mattern at ETH Zurich in 2009 and graduated in Telecommunication Engineering (with honors) from the Sapienza University of Rome, Italy, in May 2004.

Assistant professors

Gabriele Bavota
Gabriele Bavota received (cum laude) the Laurea in Computer Science from the University of Salerno (Italy). He received the PhD in Computer Science from the University of Salerno in 2013. From January 2013 to October 2014 he has been a research fellow at the University of Sannio. From November 2014 to August 2016 he was Assistant Professor at Free University of Bolzano-Bozen. He joined the Faculty of Informatics of USI as an Assistant Professor in September 2016. His research interests include software maintenance and evolution and empirical software engineering. On these topics, he authored over 100 papers appeared in international journals and conferences. He is the recipient of the 2018 ACM Sigsoft Early Career Researcher Award for outstanding contributions in the area of software engineering as an early career investigator.
gabriele.bavota@usi.ch

Piotr Didyk
Piotr Didyk is an assistant professor at Università della Svizzera italiana (USI) where he leads a group on Perception, Display, and Fabrication. He received a Ph.D. in computer science from Saarland University and Max Planck Institute for Informatics in 2012. Later, he spent two years at Massachusetts Insitute of Technology as a postdoctoral associate. In 2014, he became an independent research group leader at Cluster of Excellence Multimodal Computing and Interaction, Saarland University. At that time, he was also appointed as a Senior Researcher at Max Planck Institute for Informatics. He moved to Lugano in spring 2018. His research interests include human perception, computer graphics, new display technologies, image/video processing, 3D printing, and computational fabrication. His primary goal is to combine a deep understanding of human perception with computation and hardware to improve display quality as well as 3D fabrication techniques.
piotr.didyk@usi.ch

Michael Multerer
Michael Multerer is a Mathematician who works on uncertainty quantification for partial differential equations. He is interested in developing and implementing efficient algorithms to address the resulting high dimensional problems. Michael received his PhD in Mathematics at the University of Basel in 2014. Afterwards, he had several postdoc positions at the University of Basel, the EPF Lausanne and the ETH Zurich. In 2018, he started as assistant professor at the Institute of Computational Science at the USI Lugano.
michael.multerer@usi.ch

Nate Nystrom
Nate Nystrom is an Assistant Professor in the Faculty of Informatics at the Università della Svizzera italiana, Switzerland. His research interests include programming languages, compilers, tools, and methodologies for constructing safe, secure, and efficient systems. Prior to his current position, he was an Assistant Professor in the Department of Computer Science and Engineering at the University of Texas at Arlington and was a postdoctoral researcher at IBM T.J. Watson Research Center in Hawthorne, NY. Nystrom received his Ph.D. in Computer Science from Cornell University in 2007. He also holds B.S. and M.S. degrees in Computer Science from Purdue University and an M.S. in Computer Science from Cornell.
nate.nystrom@usi.ch

Robert Soulé
Robert Soulé is an assistant professor at the Università della Svizzera italiana (USI). His research interests are in distributed systems, networking, and applied programming languages. Prior to joining USI, he was a postdoctoral associate at Cornell University. He received his PhD from New York University in 2012, and his BA from Brown University in 1999. For two years, he was a research co-op in the Data Intensive Systems and Analytics Group at IBM T. J. Watson Research Center. He is the recipient of an IBM Invention Plateau Award, a Google Faculty Research Award, and Best Paper Awards at ACM DEBS 2012 and USENIX NSDI 2018.
robert.soule@usi.ch

Adjunct professors

Luca Maria Gambardella
Luca Maria Gambardella is director of IDSIA, Dalle Molle Institute for Artificial Intelligence, a joint research institute between USI and SUPSI in Manno. He is Professor at SUPSI (Algorithms course) and Adjunct Professor at the Informatics Faculty at USI (Artificial Intelligence course). He is leading the Intelligent Systems Master Research Unit at SUPSI. His major research interests are in the area of optimization, swarm robotics and multi-agent learning, applied to academic and real-world problems. In particular, he has invented influential and frequently cited Ant Colony Optimization algorithms for combinatorial optimization problems. He is leading projects for the Swiss NSF, Swiss CTI, EU Commission and Industries. To date (Feb 2016) he has published more than 275 publications. According to Google scholar his h-index is 57 and his articles have received more than 34200 citations. He is president of the Swiss Operations Research Society and it was president of the Euro Excellence In Practice Award 2015.
luca.gambardella@usi.ch

Post-docs and researchers

Alessio Benavoli
Marco D'Ambros
Daniele Di Marino
Davide Eynard
Alessandro Facchini
Marco Favino
Alessandro Giusti
Monica Landoni
Vittorio Limongelli
Andrea Mocci
Maria Giuseppina Chiara Nestola
Luca Ponzanelli
Simone Pezzuto
Mauro Prevostini
Paulo Rauber
Marco Zaffalon

External Lecturers

Giovanni Barone Adesi
Nikolaus Beck
Marco Brambilla
Christian Cachin
Stefano Calciolari
Cinzia Cappiello
Francesco Ciabuschi
Gianluca Colombo
Roberta De Sanctis
Daniel Florian
Francesco Franzoni
Matteo Silvano Ermanno Garzoli
Michael Gibbert
Paulo Gonçalves
Jim Kauffman
Benedetto Lepori
Loriano Mancini
Andreina Mandelli
Dirk Martignoni
Davide Martinenghi
Antonio Mele
Monica Mendini
Luca Mottola
Natasha Vijay Munshi
Francesca Pallotti
Léna Pellandini-Simány
Alberto Plazzi
Matteo Prato
Sebastian Reich
Paolo Rossetti
Luca Solari
Luca Visconti

Publisher	USI Università della Svizzera italiana
Production	Faculty of Informatics
Grid design	CCRZ
Layout	Servizio grafica
Print	Salvioni arti grafiche, Bellinzona
Number of copies	200
Date	September 2018
Orders	USI Università della Svizzera italiana Via Buffi 13 CH-6900 Lugano, Switzerland 41 58 666 00 00 +41 58 666 46 47 +info@usi.ch www.usi.ch

Università
della
Svizzera
italiana

**Faculty
of
Informatics**

**Plan of studies
3-5-8**

2018/19