

Department of Computer and Information Science
Linköping

**International
Graduate School in
Computer and Information Science**

**Courses 98/99
Including the Industry Research School**

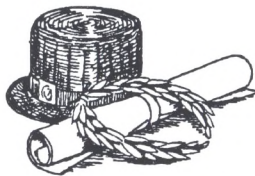
International Graduate School in Computer Science

The Computer Science Graduate School is aimed at both industry and the academic world and covers the following subject areas:

- * Computer Science
- * Computer Systems
- * Information Science and Media
- * Computational Linguistics
- * Economic Information systems
- * Information Systems Development
- * Engineering Information Systems

The Department runs an Industry Research School, as a special programme funded by The Foundation for Knowledge and Competence Development, and participates in the graduate schools Excellence Center in Computer and Systems Engineering (ECSEL), Graduate School for Human-Machine Interaction (HMI) and International Graduate School of Management and Industrial Engineering (IMIE). The research environment is strongly influenced by cooperation at both departmental and international levels and the department is regularly visited by guest professors and graduate students from international study programs. The aims of the graduate school are the following:

- The graduate school emphasizes the value of an integrated course of education in an area of importance for Swedish industry. The aim is to provide the student with broad competence. On completion of studies the student will have deep insights into his or her area of study as well as being well-oriented in the state of the art in related fields.
- The department has 18 research laboratories and all graduate students belong to one of them. This provides an environment where the student, supported by advisors, formulates and produces his or her thesis as part of the requirements.
- In addition to a main advisor each graduate student has two deputy advisors. The advisory group can provide the student with a wider range of support than is possible with just one advisor.
- The course-work pursued is of central importance in gaining broad competence. The department offers a well-established program of about 30 courses per year. These are often of an interdisciplinary character, thus the range is not limited to the student's particular lab, but is of relevance to the department as a whole. In addition to courses of a more "technical" nature, others are given in research methodology and scientific writing. Each laboratory also runs courses specific to its range of interests.
- As a consequence the study program promotes communication between students pursuing different interests. Seminar series, graduate student conferences, information and assessment meetings also stimulate collaboration. Methods of continually assessing progress and results and proposing improvements to achieve this end are considered essential.
- In addition to traditional graduate studies the department's aims have for many years included the further education of teachers and lecturers at regional University Colleges, as well as continuing education for applicants from industry.



Further information concerning the Graduate School of Computer and
Information Science can be obtained from

Lillemor Wallgren

Director of Graduate Studies Administration

Department of Computer and Information Science

Linköpings universitet, S-581 83 Linköping Sweden

Phone: +46 13281480 (281000) • Telefax +46 13142231 • Internet: lew@IDA.LIU.SE

Department of Computer and
Information Science



LINKÖPINGS UNIVERSITET

www.ida.liu.se

Contents

Schedule Fall 1998

<i>Graduate School of Computer and Information Science Graduate Program</i>	1
---------------------------------------------------------------------------------	---

<i>Industry Research School Applied IT and Software Engineering</i>	3
---------------------------------------------------------------------	---

General Graduate Courses Fall 1998:

Activity Theory/ Toomas Timpka	5
Computer Security & Systems Controls/ Nahid Shahmehri	6
Emerging Technologies in Accounting and Auditing/ Barbro Back	8
Functional Programming/ Henrik Nilsson	9
Fundamentals in Human-Computer Interaction(HMI)/ David Carr	11
Industrial Strength Formal Methods/ Anders Törne	12
Informationsteknik och management/Thomas Falk	13
Introduction to Research Methodology in Computer Science/ Sture Hägglund	14
Presentationsteknik/ Ingela Dellby	15
Programming 3D Graphics and Virtual Reality (ECSEL)/ Peter Fritzson	16
Real-Time Systems/ Anders Törne	18
Selected Notions in the Theory of Computing (ECSEL)/ Jan Maluszynski	19
Software Reliability Engineering (ECSEL)/ Mary Helander	21

General Graduate Courses Spring 1999:

Advances in Database System Technology(ECSEL)/ Tore Risch	23
Advanced Issues in Computer Architecture/ Petru Eles	25
Aspekter av vetenskapligt skrivande/ Ulf Nilsson	27
Data Mining and Knowledge Discovery/ Ankica Babic	29
Distributed Algorithms/ Lenka Motyckova	30
Evaluation of Information Systems/ Toomas Timpka	31
Fundamentals of Modern Database Systems(ECSEL)/kontakt Nahid Shahmehri	32
Industrial Project Management (IMIE)/ Christian Berggren	34
Introduction to Description Logics/ Patrick Lambrix	36
Knowledge Management/ Nils-Göran Olve och Birger Rapp	38
Object-Oriented Development of Usable Systems, basic course (ECSEL)/kontakt Peter Fritzson	39
Principles of Knowledge Representation(ECSEL)/ Erik Sandewall	41
Principles of Programming Languages and Environments (ECSEL) / Peter Fritzson	42
Qualitative Approaches to HCI (HMI)/ Jim Nyce	44

These courses will be given if enough participants show interest

Aktuella redovisningsproblem, Fall 1998-Spring 1999	45
Ekonomisk brottslighet, Fall 1998	46
GIS in Business and Service Planning, Spring 1999	47
GIS/GeoInformatik, Spring 1999	49
Natural Language Interfaces (HMI), Spring 1999	50
IT-ekonomi och informationsekonomi, Spring 1999	51
Seminar in User Interface Software Engineering (HMI), Spring 1999	52
Speech Technology (HMI), Spring 1999	53
Utredningsmetodik och kvantitativa metoder, Fall 1998- Spring 1999	54

Graduate Course Programs 98/99 at ECSEL, HMI, IMIE

55

Recommended Master Courses

57

Laboratory-oriented Courses and Activities

59

Research Organization and Laboratories

61

Faculty

66

Graduate Courses Fall 1998

Course	Course literature	Author	Aug	Sept	Oct	Nov	Dec	Day	Time	Place
Activity Theory Toomas Timpka 4 p	Activity Theory in Human-Computer Interaction	Nardi B.			Starts Oct 7			Wednesday	13-16	Elogen, Block E
Computer Security & Systems Controls Nahid Shahmehri 3-4 p	1. Information Technology Security: Managing Challenges and Creating Opportunities. 2. Computer Security Management, Course Technology	1. Fink D. 2. Forcht K.				Starts Nov 11		Wednesday Friday	9-12 9-12	Belöningen, Block B
Emerging Technologies in Accounting and Auditing Barbro Back 5 p	To be decided later			Starts week 35	week 41 week 42	Ends week 49				
Functional Programming Henrik Nilsson 5-7 p	Advanced Functional Programming. Two compendiums	Jeuring J, Meijer E.		Starts Sept 2				Wednesday Friday	10-12 10-12	Gasellen, Block G
**Fundamentals in Human-Computer Interaction David Carr 5 p	1. Designing the User Interface 2. Fundamentals of Behavioral Statistics	1. Shneiderman B. 2. Runyon R. & Haber A.		Starts Sept 3				Thursday	13-15	Estraden, Block E
Industrial Strength Formal Methods Mike Hinchey 3+2 p	Industrial Strength Formal Methods	Hinchey M. Bowen J.			Starts Oct 5			Intensive course All week-days	9-12	Belöningen, Block B
Informationsteknik och management Thomas Falk 5 p			Separate Schedule							
Introduction to Research Methodology in Computer Science Sture Häggglund 3 + 2 p				Starts Sept 29				Friday	13-15	Eliten, Block E
Presentationsteknik Ingela Dellby 3 p	Articles.			Start Sept 1				Tuesday	9-12	Belöningen, Block B, but Oct 6, 13 Betinget Block B
*Programming 3D Graphics and Virtual Reality Peter Fritzson 3 p	1. Virtual Reality Systems 2. OpenGL Programming Guide 3. Article: Interactive Multiuser VEs in the DIVE system	1. Vince J. 2. 3. Hagsand O.				Starts Nov 3 Nov 17, 24		Tuesday Tuesday	13-17 9-12,13-17	Gasellen, Block G
Real-Time Systems Anders Törne 3 + 2 p	Advances in Real-Time Systems	Sang H. Son				Starts Nov 17		Tuesday	9-12	Belöningen, Block B
Selected Notions in the Theory of Computing Jan Maluszynski 6 p				Starts Sept 17				Thursday	9-12	Elogen, Block E, but Sept 24, Oct 22, Dec 17, Eliten, Block E
*Software Reliability Engineering Mary Helander 4 p	1. Software Reliability Modeling. 2. Software Reliability Engineered Testing Practice 3. Selected papers	1. Xie M 2. Musa J.D.			Starts Oct 12			Monday	9-12	Gasellen, Block G

*ECSEL

**HMI

Graduate School of Computer and Information Science

Graduate Programme

General Information about Graduate Studies in Computer and Information Science

Graduate studies at the department consists of courses and project participation. The course programme is organized at the department level as *regular courses*, each of which is given approximately every second or third year (if possible), and *occasional courses* which depend on the profile and interests of current faculty and visiting scientists. The programme covers the areas: Computer Science, Computer Systems, Information Systems and Media, Economic Information Systems, Computational Linguistics, Engineering Information Systems and Information Systems Development.

A special study programme for industry-based graduate students is available in the area of software engineering. This Industry Research School is funded by The Foundation for Knowledge and Competence Development and by participating companies.

The Department also participates in three special graduate schools aiming for interdisciplinary studies preparing also for a career outside the university, with funding from the Foundation for Strategic Research. HMI, Human Machine Interaction started in 1997 and its goal is to improve Swedish competence by educating specialists in HMI. It is a cooperation between Linköping (IDA, IKP, Tema-K) and Stockholm (NADA, DSV). ECSEL, Excellence Center in Computer Science and Systems Engineering, started in 1996 in cooperation with primarily the Department of Electrical Engineering. IMIE, International Graduate School of Management and Industrial Engineering, has been in operation some years with contributions from the subject area Economic Information Systems in our department. Graduate students in these schools belong to research groups in the home department, but follow a special study programme.

About 100 Ph. D. students participate in the graduate programme, and may choose among about 30 courses given each year. The courses and seminars are normally given in English (unless all participants are fluent in Swedish).

The programme leads to one of the following degrees:

Licentiate of technology or philosophy. The requirements include 40 points (one point equivalent to one week full time studies) of completed courses and 40 points thesis work. For a licentiate of technology, a master of engineering ('civilingenjör', 4.5 years of study) is normally assumed as a prerequisite.

Doctor of technology or philosophy. The requirements are 80 points courses and 80 points of thesis work. Most of the Ph.D. students take the licentiate degree as an integral part of their doctoral studies.

The Research Committee, headed by Lars Ahrenberg, is responsible for the organization and implementation of the graduate programme. The members of the research committee are mainly senior researchers, but there are also representatives from the Ph.D. students, and from the research administration. As an executive, there is one director of graduate studies (Nahid Shahmehri). However, most of the administration and organization rests upon the director of graduate studies administration (Lillemor Wallgren). Most graduate students are employed by the department, full time. They assist in undergraduate courses and other internal assignments of the laboratories, up to about 15-20% of their time. The rest of the time is spent on courses and thesis project.

Graduate School of Computer and Information Science Graduate Programme

Graduate Programme in Computer and Information Science 1998/1999

This program contains the following types of courses:

- Graduate Courses given in the department
- Graduate Courses at ECSEL, HMI, IMIE
- Recommended Master Courses
- Course information for the Industry Research School
- Laboratory-Oriented Courses and Activities

It also includes presentations of

- Research Organization
- Faculty

In addition to the graduate study courses given in the Department of Computer Science, Ph.D. students may also take courses from other departments, in particular courses from the special graduate schools ECSEL, HMI, IMIE. These courses will be announced in this program together with web-addresses for further information.

The following activities are strongly recommended:

Main seminar series on Tuesdays at 13.15.

The seminars are announced by e-mail, in the IDA-Kuriren, and occasionally by special announcement. They are usually given in Estraden, E-building, 1st floor or Belöningen, B-building, 1st floor.

Departmental coffee-breaks on Tuesdays (IDA-fika)

Current information, short presentations of new arrivals and visitors in the department, descriptions of trips and conferences etc. are given every Tuesday at 12.30 in the coffee area, E-building, 1st floor. Alternatively in the coffee area in the G Building, bottom floor.

Further information concerning the contents of this program can be obtained from Lillemor Wallgren, tel 013 /28 14 80, Nahid Shahmehri, 013/28 20 66, Britt-Inger Karlsson, tel 013/28 17 06 or for a particular course from the person responsible for the course.

Linköping, June 25, 1998

Lillemor Wallgren
Department of Computer and Information Science
Linköping University
S-581 83 Linköping.
Phone: 013-281480, Fax: 013-142231, E-mail: lew@ida.liu.se

Industry Research School Applied IT and Software Engineering

General Information about Graduate Studies in The Industry Research School

The special industry research school programme within the graduate school is sponsored by the KK Foundation and a number of companies. It offers

- an industry-related, high-quality doctoral programme which is based on and utilizes the special IT competence at Linköping University.
- an effective organization coordinated with the regular graduate study programme and other strategic research efforts.
- a market-oriented and individually designed research education with extensive advisor capacity, leading to an exam as PhD, Licentiat or Master of Science.

Within the programme, twenty industry doctoral students are trained each year in Software Engineering, Computer Science and Applied Information Technology. The subject areas covered by the school focus on a scientifically based, engineering approaches to the design, development and maintenance of software systems in an industrial scale, as well as on methods and tools supporting software processes. There is a special emphasis on very large systems and their interaction with the individuals and organizations involved. Applications may concern real-time systems, net-based information systems, user interfaces, IT services, etc.

Who can become an industry doctoral student?

An industry doctoral student is supposed to have

- a research orientation belonging to the programme areas of the industry research school and fulfill the formal requirements necessary to be accepted as a doctoral student within one of the academic subjects affiliated with the school.
- a well-defined connection to a company, which contributes with resources according to a formal contract.
- an approved application to the industry research school, an assigned scientific supervisor and an individual study plan, against which an annual follow-up of results is performed.

As applicants we welcome experienced professionals from industry as well as newly examined undergraduate students. As an industry doctoral student, you may be employed either by the university or by a company.

Which companies participate?

Initially participating companies include Ericsson, Telia, CelsiusTech, SoftLab, Sectra, SKF, Idonex, Focal Point and others, but new industrial partners are still welcome. Participating companies are expected to contribute a contact person and normally also a financial contribution in the order of SEK 250 each year, either in the form of man hours or as a cash contribution. The company is required to sign a formal contract for each sponsored doctoral student.

Industry Research School Applied IT and Software Engineering

The Industry Research School 1998/99

Activities in the Industry Research School started in 1997 and doctoral students are recruited continuously. Currently the following programme areas are covered.

Software Technology

Contact persons: Prof Peter Fritzson, Dr. Bengt Lennartsson

This area studies issues related to software architecture and support environments for software development. Current doctoral projects treat for instance software architecture for information networks, a conceptual framework for software design, and software support for product design and simulation. Among companies active in this area are found Ericsson Radio, Ericsson UAB, Idonex and SKF.

Software Methodology

Contact persons: Acting prof Mary Helander, acting prof Mariam Kamkar

This area studies issues related to software quality and efficiency in systems development, with a particular emphasis on processes and methods. Current doctoral projects deal with methods for software testing, requirements engineering and usability engineering of telecom services. Active companies include Ericsson Radio, Focal Point, and Telia.

Applied IT

Contact persons: Prof Birger Rapp, prof Sture Hägglund.

This area studies issues related to IT management and strategies, software system maintenance and electronic commerce. Current doctoral projects deal with, for instance, executives' use of communication technology, effects of IT investments, IT strategies and electronic business communication. Active companies include Cepro, SYSteam, IVA and Telia.

Other research areas

Examples of other areas of interest for industry doctoral students are database technology (prof Tore Risch), infra informatics/internet (doc Patrick Doherty, doc Henrik Eriksson, prof Nahid Shahmehri), usability engineering (prof Kjell Ohlsson), real-time software (doc Ulf Nilsson, dr. Anders Törne) language technology and multimodal interaction (prof Lars Ahrenberg, dr Arne Jönsson). hardware/software codesign (prof Zebo Peng), etc.

Courses.

There are some courses arranged especially for the industry doctoral students, for example the course in industrial project management and some concentrated courses in the Software Tutorials series, SOFT. The most recent offering was SOFT-20, More Reliable Faster, Cheaper Testing through Software Reliability Engineering with John D Musa in March 1998. In general industry doctoral students are taking courses from the regular graduate study programme.

Information about the Industry Research School can be obtained through the manager:

Sture Hägglund, Computer Science Dept., Linköping University, 581 83 LINKÖPING
Phone: 013 - 281431, Fax: 013 - 142231, Email: StuHa@ida.liu.se

Activity Theory

Recommended for:

Students in Informations Systems development and Informatics.

Lectures:

16 h

The course last ran:

1996.

Goals:

To communicate the theoretical background of Activity Theory and its relevance for the development of information systems.

Prerequisites:

Basic courses in Information Systems Development and Informatics.

Organization:

Lectures cover the fundamental theory. Homeworks will be discussed during group works.

Contents:

Theoretical overview. Application of Activity Theory in Information systems development.

Literature:

Nardi B (ed). : Activity Theory in Human-Computer interaction. Cambridge, MIT Press, 1997.

Compilation of scientific papers.

Examiner:

Toomas Timpka.

Schedule:

October-December 1998.

Examination:

Attendance and term paper.

Credit:

4 credits.

Computer Security & Systems Controls

Recommended for:

PhD Students at IDA with different backgrounds.

Lectures:

20 h

The course last ran:

New course.

Goals:

Upon Completion of this subject, Students will have :

- an appreciation on how basic audit techniques can be adapted to control EDP environments.
- an understanding of the importance of enterprise security approaches to security implementation and management.
- an exposure on how to recognise security and control weaknesses in EDP systems.
- an understanding of basic risks, performance of general risk assessment exercises and recommendation of management strategies.
- an understanding and explanation of measures available to secure physical locations, computer sites, data and personnel.
- an understanding of the common terminology used within the security industry.

A key objective of this subject is to prepare the students for a holistic view of enterprise security in the context of business strategic views.

Prerequisites:

A good understanding of the Information Systems area as well as some exposure to management issues associated with computer environments.

Organization:

The course will be delivered in an on-campus mode via a series of lectures, videos, students' presentations, case studies, individual research paper as well as a real life risk analysis field exercise.

Contents:

- Introduction to computer security, framework of the course, terminology
- Security policy development and implementation, Current International Standards
- Risk management and analysis
- Physical security issues
- People/Personnel security issues
- Technical security issues
- EDP Auditing/Data Quality
- Internet Security (secure transactions)

- Cryptography
- Disaster planning/recovery
- Social/Ethical Issues in computer security
- (A possible Industry forum on current hot issues at the time)

Literature:

- Fink D., Information Technology Security: Managing Challenges and Creating Opportunities, CCH Australia Limited, Sydney, 1997. ISBN: 1-86264-942-1.
- Forcht K., Computer Security Management, Course Technology, Thomson Publishing Company, ITP, 1994.

Teachers:

The course is taught by Maurice Abi-Raad (from RMIT).

Examiner:

Nahid Shahmehri.

Schedule:

November - December 98.

Examination:

Individual research paper class presentation (depending on the number of students) syndicate group risk analysis exercise Open book case study test at the end.

Credit:

3-4 credits.

Comments:

Intensive course.

Emerging Technologies in Accounting and Auditing

Organization:

Articles according to teachers instructions use of ES, NN and GA-programs.

Contents:

The course focuses on computational intelligence in accounting and auditing. Computational intelligence refers to several new computing paradigms within computer science. It encompasses the methods of expert systems (ES), neural networks (NN) and genetic algorithms (GA). Intelligence within expert systems is embodied in their knowledge base, for example in the form of rules and facts. Expert systems have proved their worth as assisting systems in diagnostics, planning and scheduling tasks in different disciplines. Neural network receive their intelligence through a learning procedure in which the network learns by examples. They have proved their worth in several real-world problems where classification, pattern recognition, and forecasting is needed. Generic algorithms refer to a family of computational models inspired by evolution. They receive their intelligence through selection and reproduction. Genetic algorithms have been successfully used in many difficult optimization problems. They can also be used as aid in building the rule base of an expert system and to find optimal neural networks. The course deals with the following central problems within accounting and auditing:

- Effective and efficient analytic review processes in auditing
- Analyses of financial performance for benchmarking purposes
- Predictions of corporate bankruptcies
- Datamining accounting numbers using expert systems, neural networks and genetic algorithms as support aid.

Literature:

To be decided later.

Teachers:

Barbro Back.

Examiner:

Barbro Back.

Schedule:

September - December 98.

Credit:

5 credits.

Functional Programming

Recommended for:

Graduate students in Computer Science.

Lectures:

20 h

The course last ran:

Spring 95.

Goals:

The course aims at giving the participants:

- An introduction to modern functional languages, what types there are, their main characteristics, particular features of interest, and their respective advantages and disadvantages.
- Proficiency in programming in a modern lazy functional language.
- Knowledge of some advanced, purely functional programming techniques and data structures.
- An idea of the state-of-the-art and current trends within the field.

Prerequisites:

A working knowledge of Lisp/Scheme or similar language, e.g. from TDDB92/93 Programming in Incremental Systems.

In case you do not have the required background but still want to participate, we can provide you with reading directions and possibly some exercises which will give you the required background provided you do the reading well in advance of the course start.

Organization:

Lectures, assignments, presentations, project.

The exact format depends on the number of participants.

Contents:

(Preliminary)

- Modern functional languages.
- Some theoretical background.
- Lazy functional programming.
- Infinite data structures.
- Graph reduction.
- Reasoning about functional programs.
- Polymorphic type systems and type inference.
- Algebraic data types.
- Type and constructor classes.
- Functional data structures.

- Monads.
- Functional I/O; Fudgets (a functional GUI).
- Functional languages and state.
- Debugging & performance debugging.

Literature:

Johan Jeuring, Erik Meijer (Eds.): Advanced Functional Programming, Springer Verlag, ISBN 3-540-59451-5, 1995.

Two compendiums:

- Supplementary articles.
- Haskell (tutorial, language report, HBC manual).

Teachers:

Henrik Nilsson.

Examiner:

Henrik Nilsson.

Schedule:

September - October 98.

Examination:

- Compulsory assignments.
- Presentation and discussion of solutions to the assignments.
- A project (1 to 3 points)

(The exact examination forms are somewhat dependent on the number of participants.)

Credit:

5 to 7 points depending on the size of the project.

Fundamentals in Human-Computer Interaction

Recommended for:

All HMI and interested graduate students.

Lectures:

30 h

The course last ran:

New course.

Goals:

To give students a broad overview of the user interface development process. To provide experience with usability evaluations.

Prerequisites:

A basic course in statistics and an advanced programming course.

Organization:

Lectures with group projects.

Contents:

The course covers human factors issues in the development of software, use of database systems, and design of user interfaces for interactive systems. This includes theories of man-machine interfaces, task analysis models, usability studies, and controlled experimentation, as well as software engineering with user interface development environments. Topics include command languages, menus, forms, direct manipulation, graphical user interfaces, computer supported cooperative work, information search and visualization, input/output devices, and display design.

Literature:

- B. Shneiderman, Designing the User Interface, 3e, Addison-Wesley, 1997, ISBN 0-201-69497-2.
- Recommended to review statistics: R. Runyon & A. Haber, Fundamentals of Behavioral Statistics, 8th Edition, McGraw-Hill (1996).

Teachers:

David Carr.

Examiner:

David Carr.

Schedule:

September - December 98.

Examination:

Students will do a group project which includes implementing a user interface using a UIMS such as Visual Basic and conducting an evaluation of that interface.

Credit:

5 credits.

Industrial Strength Formal Methods

Recommended for:

PhD students in Computer Science and Systems. Also for ECSEL and ARTES students.

Lectures:

24 h

The course last ran:

New course.

Prerequisites:

Basic discrete mathematics, Software engineering foundations.

Contents:

- Formal specification Design in Z, CSP, OBJ
- Refinement
- Proof correctness
- Tool support
- Theorem proving in Nqthm
- Code generation
- Industrial case studies

Literature:

Industrial Strength Formal Methods, ed by Michel Hinchey and Jonathan Bowen, Academic Press, 1998.

Teachers:

Mike Hinchey.

Examiner:

Anders Törne.

Schedule:

Oct - Nov 98.

Examination:

To be decided.

Credit:

3 + 2 credits.

Informationsteknik och management

Recommended for:

Forskarstuderande inom IDA, IMIE och EKI.

Lectures:

36 h

The course last ran:

Höstterminen 97.

Goals:

Kursens mål är att förmedla en företagsekonomisk helhetssyn på IT som strategisk resurs i näringsliv och förvaltning.

Prerequisites:

Inga.

Organization:

Kursen ges under höstterminen 1998. Den består av föreläsningar, litteraturstudie, seminarier, skriftlig bokrecension samt en av varje doktorand författad litteraturbaserad uppsats (ca 15 sidor) inom ett ämne som väljs i samråd med examinator.

Contents:

Den moderna organisationen har blivit allt mer informationsbaserad och starkt informationsberoende. IT har blivit en förutsättning för företagets konkurrensförmåga och för en effektiv offentlig förvaltning. IT är en strategisk resurs och information ett viktigt konkurrensmedel. I allt större utsträckning ingår IT i nya affärsideer, produkter, tjänster och produktionsprocesser. IT bidrar på ett avgörande sätt i modern industriproduktion, produktutveckling, marknadsföring, varudistribution och logistik. IT är ett effektivt verktyg för organisationsförändringar, nya arbetssätt och affärsprocesser.

Literature:

Beslutas senare.

Teachers:

Thomas Falk.

Examiner:

Thomas Falk.

Schedule:

Oktober-December 1998.

Examination:

Skriftlig bokrecension, godkänd uppsats samt opposition.

Credit:

5 poäng.

Introduction to Research Methodology in Computer Science

Recommended for:

New graduate students. Special study groups may be arranged for students with common interests, such as students in the Industry Research School, HMI Research School, etc.

Lectures: 16

The course last ran:

Fall 98.

Goals:

Prerequisites:

None.

Organization:

Lectures and seminars. Optional study groups for extra course credit.

Contents:

Computing as a discipline. Introduction to the philosophy of science. Scientific writing, publication and information retrieval. Science and technology, methodological issues. The PhD study process. Aspects of ethics and quality control in scientific work. Research funding and politics.

Literature:

Chalmers: "What is this thing called Science?"

Teachers:

Sture Hägglund and invited guests.

Examiner:

Sture Hägglund.

Schedule:

September - November 1997.

Examination:

Written examination and seminar activity. Study group report for 2 extra credit points.

Credit:

3 + 2 credits.

Presentationsteknik

Recommended for:

Alla doktorander.

Lectures:

30 h

The course last ran:

New course.

Goals:

- Öka säkerheten att framträda inför grupper och på internationella konferenser
- Möjlighet att prova olika presentationstekniker
- Lära sig olika anteckningstekniker
- Öka det engelska ordförrådet

Prerequisites:

Inga.

Organization:

Föreläsningar, seminarier, interaktion deltagare/lärare.

Contents:

Presentationsteknik, pedagogiska metoder, engelska språket.

Literature:

Artiklar och utdrag ur böcker om presentationsteknik.

Teachers:

Ingela Dellby.

Examiner:

Ingela Dellby.

Schedule:

September 98.

Examination:

Hålla föredrag och aktivt delta och utvärdera andras presentationer.

Credit:

3 credits.

Comments:

Förslagsgivare: Lillemor Wallgren.

Kursen är begränsad till max 10 deltagare.

Programming 3D Graphics and Virtual Reality

Recommended for:

This course is recommended for ECSEL graduate students, and master students in computer science or systems engineering.

Lectures:

20 h

The course last ran:

Spring 97.

Goals:

The goal of the course is to give the participants knowledge about effective techniques for programming 3D graphics, animation and virtual reality applications.

Prerequisites:

Basic programming knowledge.

Organization:

Organization: Lectures and programming labs with practical programming exercises.

Contents:

- Virtual reality and Virtual environments.
- VR systems.
- 3D graphics.
- Geometric modeling and transformations.
- Hierarchical structured graphics objects.
- Parametric surface techniques.
- Rendering and texture.
- Interaction with 3D objects.
- 3D graphics standards such as OpenGL.
- Animation.
- Virtual reality and modeling.
- Physical simulation.
- Cooperative work within virtual reality environments.
- Designing virtual environments.
- Current hardware architectures for 3D and animation.

Literature:

- OpenGL Programming Guide, Addison-Wesley, 1995.
- John Vince: Virtual Reality Systems. ACM Press, 1995.
- Article: Olof Hagsand: Interactive Multiuser VEs in the DIVE system. IEEE MultiMedia, 1996.

Teachers:

Peter Fritzson (petfr@ida.liu.se) + invited speakers.

Examiner:

Peter Fritzson.

Schedule:

Fall 98.

Examination:

Written examination. Obligatory programming exercises.

Credit:

3 credits.

Real-Time Systems

Recommended for:

Phd students in Computer Science and Computer Systems.

Lectures:

24 h

The course last ran:

New course.

Prerequisites:

Processprogramming TDDA21 or corresponding - basic knowledge in finite automata, logic and operating systems.

Organization:

The course will comprise 24 hr lectures. About 6-9 of these will be presentations by students. Extra points are given for a performed project work.

Contents:

- Scheduling and Resource Management
- Distributed real time systems
- Real-Time Operating Systems
- Real Time Languages - synchronous and asynchronous
- RT-databases
- RT-system design

Literature:

Sang H. Son (ed), Advances in Real-Time Systems, Prentice Hall 1995.

Examiner:

Anders Törne.

Schedule:

November 98 - January 99.

Examination:

Home exam for 3p. Project report for extra 2p.

Credit:

3 + 2 credits.

Selected Notions in the Theory of Computing

Recommended for:

Foundational course for ECSEL.

Lectures:

38 h

The course last ran:

1997.

Goals:

The course presents selected topics in theory of computing. The basic concept in this context is a notion of discrete computation, or discrete process. At an abstract level it can be seen as a sequence of transitions between states triggered by some events. Thus, we will discuss systems consisting of states and transitions observing certain rules. Such systems turned out to be very useful, for example in modelling industrial processes, in design of digital circuits or in compiler construction. The course will survey several specific classes of the systems which are of particular importance for the applications.

Computations performed by computers follow certain algorithms described by the programs controlling the computations. The notion of algorithm is thus another basic concept in the focus of this course. It has been formalized by Alan Turing in terms of a state transition system known as Turing machine. We discuss how formalisation of the notion of algorithm allows one to show that some practically interesting problems are undecidable (i.e. there are no algorithms solving them), or intractable (there are no sufficiently efficient algorithms).

In addition we survey some commonly used data structures, like lists, trees, stacks, and some commonly used algorithms for searching and sorting. On these examples we demonstrate techniques for complexity analysis of the algorithms.

Prerequisites:

MSc in a non-computer-science area and some programming experience.

Organization:

Lectures and discussion sessions.

Contents:

The course consists of the following parts.

1. Abstract Automata and Formal Languages. This part is motivated by applications such as modelling of discrete event systems or compiler construction. It presents the classical transition systems: Finite Automata (deterministic and nondeterministic) and Push-Down Automata, and the corresponding languages: regular and context-free. These languages are also characterized by grammars, commonly used as a language specification formalism.

2. Petri Nets and the modelling of systems. Petri Nets provide a natural formalism for modelling of systems. We survey basic concepts and discuss some applications of the formalism, e.g. for design of digital circuits or for modelling of concurrent processes, and their synchronisation. We show relation to abstract automata discussed in Part 1.

3. Algorithms. We discuss the notions of algorithmic problem, algorithm and data structure. We illustrate them by examples of some commonly used algorithms on data structures. We survey

some basic notions of complexity theory and discuss their relevance for analysis of algorithms.

4. Tractability and decidability. We discuss Turing machines as formalisation of the notion of algorithm. We show its relevance for studying decidability and tractability of algorithmic problems. We present an important concept of NP-complete problem. Many practically relevant algorithmic problems, like planning and scheduling fall in this class. We give examples of such problems, among others the Constraint Satisfaction Problem

Literature:

D. Kelley Automata and Formal Languages. An Introduction, Prentice Hall, 1995.

J.L Peterson Petri Net Theory and the Modelling of Systems , Prentice Hall, 1981, Chapters 1-4. (For Part 2 of the course)

D. Harel Algorithmics, 2nd edition, Addison Weseley, 1992 Chapters 1-9. (For Parts 3 and 4 of the course)

Examiner:

Jan Maluszynski.

Schedule:

Fall 1998.

Examination:

Home assignments, presentations at the discussion session.

Credit:

6 credit points (for graduate students not having studied this material before).

Software Reliability Engineering

Recommended for:

This course is intended for students interested in reliability testing and measurement, and reliability analysis of software systems. It is also relevant to those interested in reliability analysis of integrated or embedded hardware/software systems.

Related Courses: Software Engineering for Complex Systems, Testing - Problems and Techniques.

Lectures:

The course last ran:

New course.

Goals:

The goal of this course is to develop knowledge of software reliability issues related to large scale industrial software development. Another goal of this course is to build basic skills for applying software reliability engineered practice within software development projects. Basic mathematical theory of software reliability will be covered, as well as software reliability modeling, software reliability prediction, and software reliability-engineered testing. Guest lecturers (including Bo Bergman, Professor of Quality Technology and Management at IKP) are planned.

Prerequisites:

- Undergraduate course in probabilistic modeling and applied statistics.
- Some programming experience.

Organization:

Contents:

- Overview of software reliability modeling objectives in the software development process.
- Basic concepts e.g. failures, operational profiles, failure intensity.
- Relationship to hardware reliability modeling and analysis.
- Selected software reliability models
- Development of operational profiles
- Preparing and planning for reliability testing
- Executing reliability tests and analysis of data

Literature:

- Textbook: M. Xie (1991), Software Reliability Modeling, World Scientific Press, Singapore.
- Textbook: J. D. Musa (1998), Software Reliability Engineered Testing Practice, McGraw Hill, Reading Ma.
- Selected papers from the current Software Reliability Engineering literature.

Teachers:

Mary Helander

Examiner:

Mary Helander

Schedule:

Fall 98.

Examination:

Homework problems, a course mini-project and written exam.

Credit:

4 credits.

Advances in Database System Technology

Recommended for:

All PhD students at IDA and ECSEL.

Lectures:

16 h

The course last ran:

Spring 98.

Goals:

The goal is to bring the students up-to-date on the current state-of-the-art within the database field.

The area of techniques for handling large quantities of data is rapidly expanding. Classical DBMSs could only handle well-structured tabular data. Modern DBMSs can also handle complex Object-Oriented data, and semi-structured data such as text, temporal data, and spatial data.

The purpose of the course is to present the fundamental concepts, theories and realizations of the state-of-the-art within the field of database technology, with a concentration on modern Object-Oriented (OO) database systems, Extensible Databases, Query Processing, and Active Databases.

Prerequisites:

- Undergraduate courses in computer science.
- Fundamentals of Modern Database Systems or equivalent.

Organization:

Lectures.

Contents:

Topics:

- History of DBMSs and data models
- Object-Oriented DBMSs
- Extensible and 'Object-Relational' databases
- Query processing in extensible and object-relational databases
- Versioning and configurations
- Active Rules in Databases
- Long-running activities
- Main-memory and High-Performance Databases
- Real-time Databases
- Temporal Databases
- Scientific Databases
- Spatial Databases

Literature:

- R.G.G. Catell: Object Data Management + handouts
- Distributed Articles

Distributed material:

- Basic DBMS Concepts
- Background and motivation for OODBs
- OODB Concepts
- OODB Implementation
- Query optimization

Examiner:

Tore Risch.

Schedule:

Spring 99.

Examination:**Credit:**

4 credits.

Comments:

Related Courses: The advanced topics of Heterogeneous Multi-databases and Scalable Distributed Storage Structures are taught by Witold Litwin in a separate advanced study course on Multi-Database Systems (in spring 2000).

Advanced Issues in Computer Architecture

Recommended for:

IDA Ph.D. students in computer science and computer systems. ECSEL students.

Lectures:

16 h

The course last ran:

New Course.

Goals:

To present the state of the art and several advanced issues of computer architecture. Special emphasis will be placed upon memory systems and parallel computers, such as pipelined, vector and multiple processor organizations as well as distributed computers.

Prerequisites:

Basic knowledge of computer organization and digital hardware.

Organization:

Mainly lectures given by the teachers, which will be supplemented by seminars prepared by the participants.

Contents:

- Introduction
- Memory systems
- Caches
- Virtual memory
- Interleaved memory and disk systems
- Pipelined processors
- Branch strategies and conflict detection
- Vector processors
- Multiprocessors
- Distributed systems
- Architectures for embedded applications
- Case studies

Literature:

H G. Cragon: "Memory Systems and Pipelined Processors", and selected research papers.

Teachers:

Petru Eles and Zebo Peng.

Examiner:

Petru Eles.

Schedule:

March - May 1999.

Examination:

Term paper and seminar presentation.

Credit:

3 credits.

Aspekter av Vetenskapligt Skrivande

Recommended for:

Alla doktorander inom området datavetenskap.

Lectures:

10 h

The course last ran:

Fall 1997.

Goals:

Att ge grundläggande kunskap om språk, innehåll och struktur i vetenskapliga verk.

Att öka medvetenheten om problem och möjligheter i skrivprocessen.

Att ge kännedom om typografins möjligheter att lyfta fram och tydliggöra innehållet.

Prerequisites:

Svenska.

Organization:

Teorigenomgångar i föreläsningsform och grupparbeten kring ett antal hemuppgifter.

Contents:

Introduktion. Vetenskapligt skrivande --- vad är det och hur skiljer det sig från andra former av skrivande? Vetenskaplig stil. De olika delarna i den vetenskapliga uppsatsen. Om titeln, sammanfattningens, introduktionens, huvuddelens, slutdiskussionens och referensernas roll i uppsatsen.

Skrivprocessen: Ett medel för att klarlägga det egna tänkandet och forskningsuppgiftens struktur. Problemformuleringens och dispositionens roll i skrivprocessen. Kramptillstånden.

Innehållsliga och språkliga aspekter av vetenskapligt skrivande: Olika typer av dispositioner. Stycket. Om kärnmeningar och olika typer av stycken. Meningar och sambands signaler. Lite om skiljetecken. Språkfel. Vad man bör se upp med. Att skriva på engelska.

Typografiska aspekter av vetenskapligt skrivande: Typografin och dess syfte. Form och innehåll. Typsnitt. Enhetlighet och måttlighet.

Att skriva avhandling: Avhandlingen och dess syfte. Olika typer av avhandlingar.

Literature:

Meddelas senare.

Teachers:

Ulf Nilsson.

Examiner:

Ulf Nilsson.

Schedule:

Våren 99.

Examination:

Inlämningsuppgifter, grupparbeten och närvaro.

Credit:

3 credits.

Data Mining and Knowledge Discovery

Recommended for:

It applies to other graduate schools, as well.

Lectures:

10 h

The course last ran:

Fall 97.

Goals:

To overview methodologies suitable for the data exploration supported by the examples.

Prerequisites:

None.

Organization:

Mainly lectures, seminar presentations in addition.

Contents:

Quantitative and qualitative data analyses. Relating expectations of the user domains to a corresponding level of information/knowledge engineering. Addressing questions of time granularities. Estimating the scope and performance of data mining and knowledge discovery.

Literature:

To be defined later.

Teachers:

Ankica Babic.

Examiner:

Active participation, seminar presentations.

Schedule:

To be defined.

Examination:

Ankica Babic.

Credit:

3 credits.

Distributed Algorithms

Recommended for:

Computer science graduate students interested in networking, distributed operating systems, and parallel computing.

Lectures:

30 h

The course last ran:

New course.

Goals:

To give students a broad overview of the different kinds of distributed synchronization algorithms.

Prerequisites:

A course in algorithms and a course that includes concurrency control.

Organization:

Lectures and seminars.

Contents:

Distributed algorithms solve well known parallel computing problems (mutual exclusion) as well as problems typical for a distributed environment (routing). Topics include: Models of distributed algorithms, graph traversal distributed algorithms, minimal spanning tree construction, routing, synchronization algorithms, logical clocks, mutual exclusion in distributed models, leader election, Byzantine agreement, anonymous networks, global snapshot, detection of communication deadlock, centralized, hierarchical and distributed deadlock detection algorithms, and termination detection.

Literature:

G.Tel, Introduction to Distributed Algorithms, Cambridge University Press, 1994.

Examiner:

Lenka Motyckova.

Schedule:

January - April 1999.

Examination:

Credit:

4 credits.

Evaluation of information systems

Recommended for:

Students in Information Systems development and Informatics.

Lectures:

10 h

Goals:

To communicate theoretical and practical knowledge in methods for evaluation of information systems.

Prerequisites:

Basic courses in Information Systems Development or Informatics.

Organization:

Lectures cover the fundamental theory. Group works and seminars will take place around a number of case studies. The students will study and present case studies during the seminars.

Contents:

Theoretical overview. Qualitative and quantitative evaluation methods. Evaluation design. Report of evaluations. Meta-analyses.

Literature:

Artiklar.

Examiner:

Toomas Timpka.

Schedule:

Spring 1999.

Examination:

Attendance and term paper.

Credit:

4 credits.

Fundamentals of Modern Database Systems

Recommended for:

Lectures:

20 h

This is a fundamental course to be attended by students without basic database knowledge.

The course last ran:

1998.

Goals:

With this course the student will understand how to effectively use commercial database systems.

Prerequisites:

Elementary programming knowledge, and knowledge about data structures and algorithms, corresponding to the course TDDDB57 Data Structures and Algorithms.

Organization:

This course is organized as a series of lectures, with accompanying computer based self-study exercises.

Contents:

This course covers the fundamentals of the database field, i.e. is how to use computers to store and manage large quantities of data.

The first part of the course covers how to design a database, i.e. how to model reality using the so called Entity-Relationship (ER) model and how to translate ER models into efficient representations of data in computers using a Database Management System (DBMS). In particular we study how to design and use relational databases where data is stored as tables and are retrieved and updated using the database language SQL.

The course covers how a DBMS is structured and what major facilities it provides. E.g. in order to handle concurrent access to shared databases a transaction mechanism is provided, to describe the data there is a meta-data (schema) facility, to query data there is a general query facility, to reliably store data there is a recovery subsystem, and to secure data there are authorization facilities and integrity constraints, etc.

The course also gives overviews of some important recent developments within the fast-growing field of commercial database systems, e.g. Object-Oriented Databases, Active Databases, Distributed Databases, and modern PC-based database tools such as Access and Java-Builder.

Literature:

Elmasri, Navathe: Fundamentals of Database Systems, 2nd Ed, Benjamin/Cummings, 1995.

Course Compendium for independent Computer Exercises.

Examiner:

To be decided.

Contact: Nahid Shahmehri

Schedule:

January-February 1999.

Examination:

Credit:

3 credits.

Industrial Project Management

Recommended for:

Alla.

Lectures: 8 x 4 tim+ heldagar.

The course last ran:

Våren 97.

Goals:

Doktorandkursen Industriell projektledning har tre mål.

- Att ge insikter i projektledningens såväl "hårda" (planering, styrning, kontroll), som "mjuka" sidor (team building, ledarskap, krishantering, m m).
- Att förena industriella perspektiv och erfarenheter från aktuella projekt med akademisk reflektion och analys
- Att skapa fördjupad förståelse för den egna personlighetens roll genom övningar, spel, dialog och reflektion kring ledarskap och socialt samspel.

Prerequisites:

Inga.

Organization:

Deltagarna är begränsade i antal eftersom kursen innehåller vissa moment där antalet deltagare ej får vara för stort. 80% närvaro krävs för att få poäng.

Contents:

Modern industriell projektledning handlar i hög grad om interdisciplinär, internationell verksamhet, vare sig det rör sig om leveransprojekt, utvecklings- eller upphandlingsprojekt.

Kursen kommer att bygga på deltagarnas egna aktiviteter, liksom exempelvis kursen TQM och lärande organisationer. Den skall ge förståelse av organisation, ledning och styrning av olika typer av industriella projekt och förmedla insikt om samspelet mellan projekt och företagets basorganisation, och hur detta påverkar möjligheter att nå projektmål och generera långsiktigt organisatoriskt lärande. Avsikten är dels att göra en rejäl fördjupning i litteraturen om projektledning och projektledningsproblem (inklusive de klassiska "planning disasters"), dels att ha en nära koppling till praktiken genom att ta avstamp i aktuella industriprojekt inom regionen. Här kommer vi också att vända oss till PMEX-deltagarna, och bjuda hit några av dem liksom andra aktiva projektledare.

Literature:

- Loch, D (1996) The Essentials of Project Management.
- Engwall, M (1995) Jakten på det effektiva projektet.

Teachers:

Christian Berggren and guest speakers.

Examiner:

Christian Berggren, tf professor industrial organization.

Schedule:

Preliminary week 16 to 22, 1999.

Examination:

Muntlig examination.

Credit:

5 credits.

Comments:

Are you interested in such a course? Make a preliminary registration for it.

The course might be given in Swedish.

Introduction to Description Logics

Recommended for:

Graduate Students. (Fördjupningskurs C4-C3)

Lectures:

24 h

The course last ran:

1995/96.

Goals:

Description logics (or terminological logics or concept languages) are computationally attractive limited subsets of first-order logic.

They are used for representing concepts and individuals in a given world and then drawing logical inferences from the given information.

The inferences center around, but are not limited to, classification. There are a number of implemented description logic systems which are used for building a variety of applications including software management systems, planning systems, telephone configuration systems and natural language understanding. Description logic systems are particularly suitable for applications where classification is a useful inference. At the conclusion of this course you will have:

- a good understanding of the principles of description logics
- used a number of different description logic systems, and compared their differences
- an understanding of the central algorithms used in description logic systems, in particular subsumption and classification
- looked briefly at some areas of current development in description logics.

Prerequisites:

basic course in logic (e.g. Logic for idaites, Logik grundkurs (TDDA15)).

Organization:

lectures + seminars (prepared by the participants) + labs.

Contents:

- Introduction to the field, what description logics are, KL-ONE
- Representing knowledge in description logics
- The algorithms: classification, subsumption, other inferencing
- Completeness, Correctness, Complexity
- Revision and update of knowledge bases
- Systems: KRIS, LOOM, CLASSIC, CRACK, FaCTS
- Applications using description logics
- Connection to databases
- Integrating different kinds of reasoning with description logics

Literature:

Articles.

Teachers:

Patrick Lambrix.

Examiner:

Patrick Lambrix.

Schedule:

February 99.

Examination:

seminars, lab assignments, experience report.

Credit:

4 credits.

Knowledge Management

Recommended for:

Forskare med inriktning på hur företag anskaffar, förvaltar och utnyttjar immateriella tillgångar.

Lectures:

36 h

The course last ran:

Ny kurs.

Goals:

Gemensamt utforska innebörden av ett modebegrepp som vi tror speglar ett angeläget behov i flertalet organisationer. Kursen har sin bas i ett företagsekonomiskt perspektiv som innebär att organisationer medvetet strävar att nå vissa mål genom att utveckla och utnyttja resurser som står till dess förfogande. Om "kunskap" är en allt viktigare sådan resurs, hur kan detta ske?

Prerequisites:

Inga formella. Bakgrund i företagsekonomi kan dock i praktiken behövas för att förstå delar av resonemang och litteratur.

Organization:

Kursen omfattar ca 5 heldagar under våren 1999. De tidigare utgör en exposition av olika tankestoff som kan bidra till området; den sista (eller, vid flera deltagare, de två sista) en redovisning av individuella arbeten.

Contents:

Ämnet har i olika sammanhang anknytning till åtminstone följande områden: Kompetensutveckling och lärande; Databasarkitektur; Artificiell intelligens; Företagsledning i kunskapsföretag (ansvar, strategibeslut); Intellectuellt kapital (värdering, marknadsmekanismer, immaterialrätt); Rationellt och intuitivt i människors samverkan. Dessa kommer att belysas och kontrasteras.

Literature:

Tänkbar basbok Davenport & Prusak "Working Knowledge" (1998). Prusak "Knowledge in Organisations" (1997).

Examiner:

Nils-Göran Olve, Birger Rapp.

Schedule:

Januari - juni 99.

Examination:

Godkända arbetspapper, framlagda vid seminarier i juni 1999.

Credit:

5 poäng.

Object-Oriented Development of Usable Systems, basic course

Recommended for:

This course is specifically designed for those students who have not been exposed to Object Oriented software development.

Lectures:

Related Courses: Software Engineering for Complex Systems, Relation between Specification and Programming Languages, Object-Oriented Development of Usable Systems - Advanced Course.

The course last ran:

Note that this course is modeled after TDDDB34, but it is a graduate version specifically designed for ECSEL students who have not been exposed to Object Oriented software development.

Goals:

After finishing this course, the participants should:

- Understand the object oriented design model including its use for object oriented analysis and design of software systems,
- Understand principles for construction of changeable and extensible systems by reusable components,
- Should be able to use an object oriented programming language for implementation, and
- Should have good grasp of techniques and methods for development of systems which suit users in their application situation.

Prerequisites:

- Experience with some high level programming language.
- Knowledge about data structures and algorithms.

Organization:

Contents:

The course covers parts of object oriented system development, etc. This includes: (OOSU) basic object oriented terminology such as object, class, inheritance, etc; principles for object-oriented analysis and design; quality aspects of analysis and design; and object oriented implementation techniques. An overview of several object oriented development environments will be included.

OOD with respect to Usability will also be covered: Usability oriented design of functions and interfaces; implementation of object oriented user interfaces; and basic usability evaluation.

Topics are:

- Introduction
- Basic concepts such as object and class

- Basic object-oriented analysis
- Use-case analysis
- Advanced topics such as meta-classes and interfaces
- Object-oriented design
- Adaptable systems
- Re-use
- Object-oriented project management

Literature:

Recent textbook and handouts on OO development.

Teachers:

TBA.

Examiner:

TBA.

Contact: Peter Fritzson.

Schedule:

Spring 99.

Examination:

OO development projects.

Credit:

4 credits.

Principles of Knowledge Representation

Recommended for:

Foundational course for ECSEL.

Lectures:

12 h

Goals:

Knowledge representation is concerned with the systematic and formal description of real-world phenomena, with an emphasis on discrete-level descriptions of objects and processes with a complex structure. It relies on discrete mathematics (in the sense of elementary set theory, graphs, etc) and on formal logic as its conceptual tools.

The research area of knowledge representation has its roots in artificial intelligence research, but there is ample opportunity for interactions with several other fields, ranging from model-building in control engineering, via databases, to systems for human-machine interaction and natural language systems.

In addition, there is a core of common concepts which underlie the representational issues both in programming languages, databases, and knowledge systems.

The present course will present both the basic concepts of knowledge representation per se, and its connections to these neighboring areas.

Prerequisites:

MSc in a non-computer-science area and some programming experience.

Organization:

The course will include lectures, problem solving sessions, and computational exercises.

Contents:

The course consists of the following parts.

1. Discrete structures. A brief recapitulation of the kinds of systems that one wishes to characterize in formal knowledge representations, with examples.
2. First-order Predicate Logic. Syntax of logic formulae; their semantics; the concept of semantic entailment; inference systems.
3. Description of processes in first-order logic. Representations of time, persistence, indirect effects, delayed causation, etc.

Literature:

To be specified.

Examiner:

Erik Sandewall.

Credit:

3 credit points (for graduate students not having studied this material before).

Principles of Programming Languages and Environments

Recommended for:

Foundational course for ECSEL.

Lectures:

21 h

The course last ran:

1998.

Prerequisites:

MSc in a non-computer science area and some programming experience.

Organization:

The course is mainly based on lectures, estimated 21 hours, 3 hours per week. The students should have a quick reading of the material before each lecture to enable discussions. There will also be a few practical programming exercises, e.g. generating a small translator from specifications, started during an introductory laboratory session of 2-4 hours.

Contents:

1. What is a programming language? Abstractions in programming languages. Computational paradigms. Language definition, translation and design.
2. Language design principles such as efficiency, generality, orthogonality, and uniformity.
3. Language properties. Syntax, basic semantics. Data types, type information, type constructors, type equivalence, type conversion. Control constructs, procedure environments, activations, allocations. Abstract data types. Overloading and polymorphism. Modules and separate compilation. The mathematics of abstract data types.
4. Programming paradigms. Procedural programming, object-oriented programming languages, functional programming languages.
5. Basic semantics, such as bindings, semantic functions, scope, allocation, extent, the environments. Formal semantics of programming languages. Operational semantics, Natural Semantics, and Denotational semantics. Automatic generation of language implementations from specifications.
6. Principles and methods behind interactive and integrated programming environments. The notion of consistency. Examples of such environments. Source code configuration management and version control system

Literature:

Kenneth C. Loudon: Programming Languages, Principles and Practice. PWS Publishing Company, Boston, 1993. (ISBN0-534-93277-0)

Peter Fritzson: Specifying Practical Translators using Natural Semantics and RML. (kompendium). Peter Fritzson: System Development Environments. (kompendium).

Examiner:

Peter Fritzson.

Schedule:

Spring 1999.

Examination:

Written examination or homework problems. A few practical programming excercises. Active participation during lectures.

Credit:

3 credits

Qualitative Approaches to HCI

Goals:

The course will discuss the intellectual and pragmatic yields qualitative approaches, particularly ethnography, can have for the study of human-computer interaction. The course is intended to help students become more competent in assessing the research of others. By looking at the strengths and weakness of this body of literature, it will also help student design and argue for their own research and research agendas.

Prerequisites:

Graduate student status.

Contents:

This graduate seminar course will introduce students to qualitative approaches to HCI. As such there will be no prerequisites. The readings will start with Suchman's Plans and Situated Actions (1987) and will run chronologically through some of the HCI literature. The course would cover some of the Scandinavian and N. American approaches to HCI. The course will also treat current developments and research in HCI as development out of HCI's earlier history and paradigms. Given the breadth and depth of the HCI literature, the class will focus on the CSCW research literature. One topic that the course will discuss is participatory design. The course will look at Scandinavian (Bodker, Kensing/Simonsen) and N. American (Blomberg) approaches to participatory design as a practical research activity. In particular, the course will look at the contributions ethnography can make to participatory design.

Literature:

Selected readings.

Teachers:

James M. Nyce.

Examiner:

Schedule:

Spring 1999, intensive course.

Examination:

Class work will include assigned readings, seminar participation (2 points) and one research paper (1-3 points). Course credit may be changed.

Credit: 2 + 3 credits.

These courses will be given if enough participants

Aktuella redovisningsproblem

Recommended for:

De som deltar i forskningsprogrammet med inriktning på redovisning och revision.

Lectures:

25 h

The course last ran:

New course.

Goals:

Diskussion kring några av de redovisningsproblem som diskuteras internationellt med betoning av de principiella frågeställningar som därvid aktualiseras.

Prerequisites:

Inga formella krav.

Organization:

Föreläsningar och seminarier vid fem tillfällen om vardera ca 5 timmar.

Contents:

Preliminärt kommer bland annat följande att behandlas; immateriella tillgångar, finansiella instrument, avsättningar samt nedskrivningar.

Literature:

Material delas ut vid föreläsningarna.

Teachers:

Rolf Rundfelt.

Examiner:

Rolf Rundfelt.

Schedule:

September 98 - June 99.

Examination:

Enskilda arbeten.

Credit:

5 credits.

Ekonomisk brottslighet

Recommended for:

Forskare med inriktning mot revision, affärsrätt, statistiska kontrollfrågor eller datasäkerhet. Dessutom personer som i sitt arbete kommer i kontakt med ekobrottslighet, anställda vid exempelvis Skatteförvaltningen, Åklagarväsendet, Ekobrottsmyndigheten, Polisen eller Tullverket.

Lectures:

30 tim

The course last ran:

Ny kurs.

Goals:

Att ge en översikt över olika typer av ekobrottslighet samt uppskattningar av deras ekonomiska värde. Att ge en inblick i kontrollstrategiska frågor med användning av statistisk metodik. Betydelsen av datorsystem och Internet. Möjligheten att förhindra brottslighet medelst bättre systemutformning.

Prerequisites:

Inga formella.

Organization:

Kursen omfattar ca 5 heldagar under senvåren och början av höstterminen 1999. Den sista eller de två sista dagarna ägnas åt redovisning av enskilda arbeten.

Contents:

Översikt av området. Aktuella nya brottsformer. Skattekontrollens metodik och resultat. Tullens kontrollsystem. Brotsutredningar och lagföring. Kontrollstrategiska metoder. De stora statliga ekonomisystemen. Banksystemet. Internethandel.

Literature:

Statliga utredningar, bl a Branschsaneringsutredningen samt Internationella ekobrott. Rapporter från brottsförebyggande rådet. Riksskatteverkets rapport Skattefel och skattefusk. Årsrapporter från länsstyrelsernas samverkansorgan mot ekobrottslighet. Internationella tidskriftsartiklar om skattekontroll.

Teachers:

Leif Appelgren.

Examiner:

Leif Appelgren.

Schedule:

Höstterminen 1999.

Examination:

Godkända arbetspapper framlagda vid seminarier i september 1999.

Credit:

5 poäng

These courses will be given if enough participants

GIS in Business and service planning

Recommended for:

Students in Informatics, systems and computer sciences and subjects related to work with GIS in Business and service planning.

Lectures:

20 h

The course last ran:

New course.

Goals:

To Give an introduction to the use of Geographical data, tools and information systems in Business and service planning.

Prerequisites:

Graduate student.

Organization:

Lectures.

Contents:

- Introduction
- Datacapture in GIS
- Data minig
- Spatial statistics
- Information systems för descision support
- Practical laborations
- Report

Literature:

- Geographical Information Systems, Tor Bernhadsen.
- GIS for Business and Service Planning, Longley and Clarke.

Teachers:

Åke Sivertun.

Examiner:

Åke Sivertun/Birger Rapp.

Schedule:

March 1999.

Examination:

Written report.

Credit:

3 credits.

Comments:

The course will be given in cooperation with EIS (The Laboratory for Economical Information Systems).

These courses will be given if enough participants

GIS/GeoInformatik

Recommended for:

All PhD students in informatics, systems and computer science and other for GIS.

Lectures:

32 h

The course last ran:

New course.

Goals:

To give a deeper understanding for the theoretical and practical aspects of GIS.

Prerequisites:

Graduate student.

Organization:

Lectures.

Contents:

The students will read and present papers from the current litterature.

Topics include tool kits, model-based interface development, UI software architectures, user interface development systems, and user interface development methodologies.

Literature:

- Laurini, Robert & Thompsson, Derek (1992).
- Fundamentals of spatial information systems. the APIC series, Academic Press ca 700 sidor.
- Reference littterature: Chang S.K. & Jungert E. Projection for Image Information Retrieval and Spatial Reasoning, Academic Press London 1996.
- Worboys, M.F. "GIS: A Computing Perspective".

Teachers:

Teachers who will collaborate in the course are: Erland Jungert, Michael LeDuc, Tore Risch, Per Svensson, Åke Sivertun.

Examiner:

Åke Sivertun.

Schedule:

March 99.

Examination:

Written paper.

Credit:

5 credits.

These courses will be given if enough participants

Natural Language Interfaces

Recommended for:

Studenter med intresse för naturligt språk och/eller utveckling av gränssnitt.

Lectures:

24 tim

The course last ran:

New course.

Goals:

Kursen ger en översikt över forskningen idag och studenterna ska efter genomgången kurs veta hur naturligt språk kan användas i användargränssnitt och ha kännedom om metoder och tekniker för utformning av ett sådant gränssnitt och dess olika komponenter.

Prerequisites:

Grundkurs i lingvistik. Formella språk, datalingvistik alt. AI. Talteknologi rekommenderas.

Organization:

Seminarier.

Contents:

Metoder för automatisk analys och generering av språk (tal resp. text). Mänsklig dialog och människa-dator-dialog. Subspråk. Insamling och analys av dialogdata. Systemarkitekturer och dialoghantering i gränssnitt med naturligt språk. Designkriterier (robusthet, portabilitet, ...) och utvärderingsmetoder. Genomgång av applikationer och system.

Literature:

Artiklar och konferensbidrag.

Teachers:

Lars Ahrenberg, Arne Jönsson, Bertil Lyberg, Lena Strömbäck

Examiner:

Lars Ahrenberg.

Schedule:

Spring 1999

Examination:

To be decided

Credit:

3 + 2 credits.

These courses will be given if enough participants

IT-ekonomi och informationsekonomi

Recommended for:

EIS. Lämplig för andra intresserade med grunder i företagsekonomi enligt ovan.

Lectures:

24 h

The course last ran:

Spring 1998.

Goals:

Att förstå sambandet mellan IT och ekonomiska effekter i företag och samhälle.

Prerequisites:

Baskunskaper i redovisning, investeringskalkylering och strategi.

Organization:

Ett tiotal tvåtimmarsföreläsningar ges gemensamt för kursen TEUD inom grundutbildningen och denna kurs. Vid något av dessa tillfällen behandlas ett praktikfall.

Ca två månader senare lägger doktoranderna vid seminarier fram kortare arbetspapper kring självvalda uppgifter. Dessa avsåg 1997 och 1998 ämnena såsom:

IT-investeringskalkylering i en kommun; Hur fyra företag värderat nyttan vid EDI-införande; IT in IKEA; Värdera IT-nyttan vid distans- arbete. Arbetet kan med fördel användas för att testa tankar inför val av avhandlingsämne.

Contents:

IT:s roll ur ett strategiskt perspektiv, såväl för företag som andra verksamheter.

IT-investeringar och deras redovisning. Investeringskalkylering. Incitament i organisationer. Informations värde.

Literature:

Falk & Olve, IT som strategisk resurs (Liber 1996). Dessutom självvald litteratur.

Teachers:

Nils-Göran Olve.

Examiners:

Nils-Göran Olve.

Schedule:

March - June 99.

Examination:

Arbetsrapporter enligt ovan; medverkan i seminarier där de läggs fram.

Credit:

3 credits.

These courses will be given if enough participants

Seminar in User Interface Software Engineering

Recommended for:

All HMI and interested graduate students.

Lectures:

20 h

The course last ran:

New course.

Goals:

To develop an understanding of how user interface software is designed and developed and current research in the area.

Prerequisites:

A basic course user interface software development.

Organization:

Seminars.

Contents:

The students will read and present papers from the current literature.

Topics include tool kits, model-based interface development, UI software architectures, user interface development systems, and user interface development methodologies.

Literature:

- Selections from recent conference proceedings and journals, for example UIST,
- Transactions on Computer-Human Interaction, SIGCHI, and Design, Specification, and Verification of Interactive Systems.

Teachers:

David Carr.

Examiner:

David Carr.

Schedule:

February - April 99.

Examination:

Oral presentations plus a term paper.

Credit:

3 credits.

These courses will be given if enough participants

Speech Technology (HMI 621)

Recommended for:

Kurs gemensam för C- och D-linjen, Kognitionsvetenskapliga programmet samt doktorander.

Lectures:

Goals:

Kursen syftar till att ge grundläggande kunskap om olika tekniker för analys, igenkänning och syntes av mänskligt tal samt deras tillämpningar t.ex. i samband med systemstyrning, talkommunikation med informationssystem, handikaphjälpmedel och tolkhjälp.

Organization:

Kursen har en föreläsningsdel och en laborationsdel. Därutöver ingår ett studiebesök i Stockholm.

Contents:

Literature:

C. Schmandt: Voice Communication with Computers, Van Nostrand Reinhold, NY, 1994.

Övrigt kursmaterial:

Ett antal artiklar och konferensbidrag i form av en artikelsamling på drygt 100 sidor

Laborationshandledning.

Teachers:

Föreläsningar: Mats Blomberg, Bertil Lyberg, Rolf Carlsson, Kjell Elenius, Björn Granström, Arne Jönsson.

Examiner:

Arne Jönsson.

Schedule:

Föreläsningar

Schemalagda datorövningar

En heldag i Stockholm

Examination:

OBS! Examinationen ny för i år! *Hemtenta* + tre labbar.

Credit:

5 credits.

These courses will be given if enough participants

Utredningsmetodik och kvantitativa metoder

Recommended for:

Alla doktorander.

Lectures:

35 h

The course last ran:

Våren 1997.

Goals:

- väl insatt i modern utredningsmetodik
- väl orienterad om kvantitativa metoder
- känna till centrala begrepp och teorier inom vetenskapsteori

Prerequisites:

Inga.

Organization:

Föreläsningar, seminarier, inlämningsuppgifter, tentamina.

Contents:

Vetenskapsteori, kvalitativ undersökningsmetodik, modelltänkande och systemanalys, statistiska metoder, simulering, prognoser, utvärdering och presentationsteknik.

Literature:

Beslutas senare.

Examiner:

Birger Rapp

Schedule:

Oktober 98 - juni 99.

Examination:

Aktivt deltagande på seminarierna, godkända inlämningsuppgifter och laborationer samt skriftliga och muntliga tentamina.

Credit:

5 poäng.

Graduate course programmes 98/99 at

ECSEL, HMI, IMIE

Below you will find both a list of the courses which will be given during fall 1998 and spring 1999 by the graduate schools where the department is involved, web addresses and contact persons for the graduate schools in question.

Human Machine Interaction (HMI)

Responsible persons and web address:

Stockholm: Director of graduate studies Ann Lantz, alz@nada.kth.se

Linköping: Director of graduate studies Lena Strömbäck, lestr@ida.liu.se phone 013/28 26 76.

<http://www.ida.liu.se/hmi/>

Preliminary list of courses 98/99 in Linköping

HMI 503. Human Factors Research Methodology and Field Experimentation, 5 points, Fall 1998, Instructor: Alm

HMI 601. Biomechanics and Work Injuries, 5 points, Fall 1998, Instructor: Fang Chen & Jörgen Eklund.

HMI 602. Cognitive Systems Engineering, 5 points, Spring 1999, Instructor: Erik Hollnagel

HMI 603. Organization Management and Work Psychology, 5 points, Spring 1999, Instructor: Gunnela Westlander

HMI 604. Human-Computer Interaction, Spring 1999, Instructor: Helander

HMI 605: Human Reliability - Human Erroneous Actions in a System Context, 5 points, Fall 1998, Instructor Erik Hollnagel

HMI 621: Speech Technology, 5 points, Spring 1999, Instructor Arne Jönsson, IDA

HMI 622. Natural Language Interfaces, Spring 1999, Instructor Lars Ahrenberg, IDA

HMI 623 Fundamentals in Human-Computer Interaction, 5 points, Fall 1998, Instructor David Carr, IDA

HMI 624 - Seminar in User Interface Software Engineering, 3 points, Spring 1999, Instructor David Carr, IDA

HMI 701 Research Topics: Automation, Cognition, and the Human Interface, Fall 1998, Instructor S Dekker

HMI 721 - Qualitative Approaches to HCI, 2+3 points, Spring 1999, Instructor James M. Nyce, IDA

Preliminary list of courses 98/99 in Stockholm

HMI 651 Research Methods in Human-Computer Interaction, 5 points, Fall 1998, Instructors: Ann Lantz, Stockholm

HMI 661 Cognition, Learning, Instruction and Computers, 5 points, Fall 1998, Instructors: Robert Ramberg, Stockholm

Graduate course programmes 98/99 at

ECSEL, HMI, IMIE

International Graduate School of Management and Industrial Engineering (IMIE)

Responsible persons and web address:

Director of graduate studies Helén Anderson, helan@eki.liu.se, phone 013/282517

Administrator Lena Sjöholm, lensj@eki.liu.se, phone 013/282357

<http://www.liu.se/org/imie>

Courses Fall 98

TQM in Learning Organisations, 5-6 points, examiner Bo Bergman, IKP

Management Control, 5 points, examiner Lars Lindkvist, EKI

Strategic Change, 5 points, examiner Bo Hellgren, EKI

Informationsteknik och management, 5 points, examiner Thomas Falk, IDA

Produktionsstrategier, 3 points, examiner Jan Olhager, IPE

Manufacturing Systems Philosophy, 2,5 - 5 points, examiner Christer Johansson, IKP

Marknadsföring och logistik, 5 points, examiners Staffan Brege/Sten Wandel, EKI

Applied Multivariate Statistics, 5 points, examiner John Farley, EKI

Courses Spring 99

Ekonomisk teori och metodutveckling, 5 points, examiner Jan Lindvall, EKI

Strategic Management of Technology and Innovation, 10 points, examiner GLindström, EKI

Vetenskaplig metod, 10 points, examiners Helén Andersson/Roland Sjöström, EKI

Industriell projektledning, 5 points, examiner Christian Berggren, EKI

ECSEL - Excellence Center in Computer Science and Systems Engineering in Linköping

Responsible persons and web address:

Director of graduate studies Nahid Shahmehri, nsh@ida.liu.se, phone 013/ 28 20 66,

Administrator Kristin Wiberg, kristin@isy.liu.se, phone 013/28 57 15

<http://vir.liu.se/ecsel/Ecseltopp.html>

Recommended Master Courses

C3-C4-courses

TDDA12	System Development
TDDA14	AI Programming
TDDA16	Representation of Knowledge in AI
TDDA32	Design and Analysis of Algorithms
TDDA37	Compiler Construction
TDDA41	Logic Programming
TDDA43	Programming Theory
TDDA99	Kognitionsvetenskapliga kommunikationsmodeller
TDDB02	Software Quality
TDDB06	Advanced Programming and Interactivity on the WWW
TDDB08	Logik fördjupningskurs
TDDB09	Formell programutvecklingsmetodik
TDDB12	Concurrent Programming
TDDB13	Human-Computer Interaction
TDDB15	Computer Aided Software Engineering for Development and Maintenance
TDDB34	Object-Oriented System Development
TDDB38	Database Technology
TDDB55	Medieinformatik
TDDB61	Methodology of Program Development and Programming Development Project
TDDB66	Expert Systems-Methods and Tools
TDDB67	Distributed Systems
TDTS41	Computer Networks
TDTS51	Advanced Computer Architecture
TGTU04	Leadership

SVP-courses

HIIC72	Ledarskap
HIID62	Programvarukvalitet
HIID63	Forskningsstrategier inom informationssystem
HIID70	Design för användbarhet
HIID71	Teorier om design för användbarhet

Master's Program in Communication and Interactivity

IDA and the Department of Electrical Engineering offer a master's programme in computer science and engineering which focuses on topics in communication and human-machine interaction. Applicants are expected to have a Bachelor's degree in computer science and engineering

The program consists of three semesters. The first two semesters consist of advanced courses selected from the computer science and engineering syllabus in Linköping. Thesis work is carried out during the third semester.

More information about the program can be found at <http://www.ida.liu.se/~msc-ci/>

Laboratory-oriented Courses and Activities

Like the graduate courses laboratory-oriented courses and activities are open for all Ph.D. students at the department, but they are organized so as to have a direct link to activities in each laboratory. Course activities may be announced during the year. These are the fixed times for laboratory meetings and the laboratory activities announced by the laboratory VITS:

Laboratory	Fixed time for Lab meetings
ACTLAB	Wednesdays 13-15
ASLAB	No fixed time
ASELAB	No fixed time
CADLAB	Thursdays 13-15
EDSLAB	Thursdays 13-15
EIS	No fixed time
ESLAB	Wednesdays kl 10-12
LIBLAB	Wednesdays 13-15
IISLAB	Wednesdays 13-15
KPLAB	No fixed time
LOGPRO	Wednesdays 15-17
MDA	Wednesdays 13 - 15
NLPLAB	Wednesdays 15-17
PELAB	Thursdays 15-17
RTSLAB	Wednesdays 13-15
TASLAB	No fixed time
TOSCA	No fixed time
VITS	Some Mondays 10-12

The following activities will take place in VITS during the year of 98/99:
(All the activities will be in Swedish)

Verksamhetsdiagnos

Kursen ges i Jönköping med start september 98

Kursansvarig: Göran Goldkuhl

Information på <http://www.hj.se/~gogo/grut/vedikurs.html>

Seminarier i informationssystemteori

Seminarierna kommer att äga rum i Linköping under hela läsåret; start september 98

Kursansvariga: Karin Axelsson, Göran Goldkuhl

Strategisk planering av administrativ utveckling - fördjupning

Kursen ges i Jönköping under hösten 98

Kursansvarig: Mats-Åke Hugoson

Laboratory-oriented Courses and Activities

Avhandlingsseminarier

Seminarierna äger rum i Linköping under hela läsåret och är avsedda för de som aktivt skriver på lic/doktorsavhandling

Kursansvariga: Karin Axelsson, Stefan Cronholm, Göran Goldkuhl

Preliminärt kommer också följande kurser att ges:

Kunskapsprojektering

Kursen ges i Jönköping, start jan 99

Kursansvarig: Göran Goldkuhl

Vetenskapsteori

Kursen ges i Jönköping, start jan 99

Kursansvarig: Göran Goldkuhl

Kvalitativ analys och teoriutveckling

Kurser ges i Linköping under våren 99

Kursansvariga: Stefan Cronholm, Göran Goldkuhl

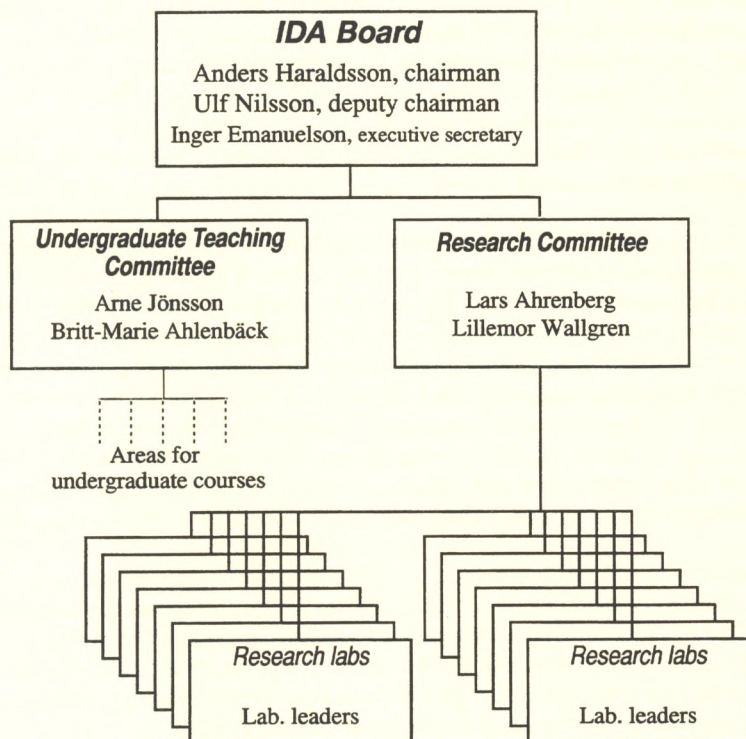
Mer information om VITS-aktiviteterna kan fås från:

Göran Goldkuhl, e-post ggo@ida.liu.se

Karin Axelsson, karax@ida.liu.se

Stefan Cronholm, stecr@ida.liu.se

Research Organization and Laboratories



The strategic and executive responsibilities for research projects and graduate education

IDA's research program has been designed to cover areas of strategic importance, both for undergraduate education as well as for the needs of society. Research in the department is currently organized in 18 research laboratories. Each such unit is characterized by its long-term commitment to develop and maintain the knowledge within a defined area, and by its long-term responsibility for individual graduate students. A short review of all research laboratories is given on the following pages.

ACTLAB - Laboratory for Complexity of Algorithms

Per-Olof Fjällström

The research of ACTLAB is focused on design and analysis of efficient algorithms and data structures for combinatorial and geometric problems; research that is highly relevant for applications such as computer-aided design and geographical information systems.

ASELAB - Applied Software Engineering Laboratory

Mary Helander

The scientific goal of ASELAB is to develop knowledge relevant for production and use of large-scale industry software. Research is carried out with various empirical methods using real-life industrial projects and software products as study material, emphasizing process improvement methods and product quality. Particular areas of study include prescriptive models of software reliability, software fault predictors, requirements engineering, robust planning profiles and software reengineering methods.

ASLAB - Application Systems Laboratory

Sture Hägglund

The research in ASLAB has several foci. One is human-computer interaction, especially usability-oriented methods for IT design, IT learning and support for the design process. Another interest is development methods and meta-level tools for knowledge engineering, supporting knowledge acquisition and reusable problem solving methods. There are also studies of cooperative expert systems, cognitive models for visual creativity, collaborative dialogues in intelligent tutoring systems and simulation for training.

CADLAB - Laboratory for Computer-Aided Design of Digital Systems

Krzysztof Kuchcinski

CADLAB concentrates its research activities on computer-aided synthesis and verification of digital systems, which are supposed to be implemented completely or partially in hardware. Such tools or methods exist already for low level design problems. Using computer science methods, we develop a design framework, where abstract design specifications are systematically translated into a concrete implementation. Our research currently concentrates on the high-level synthesis and hardware/software co-design.

EDSLAB - Laboratory for Engineering Databases and Systems

Tore Risch

EDSLAB does research on modern data-base services and applications for engineering and telecom information systems. The research areas include domain specialized database technology, object-relational mediators, distribution, integration of heterogeneous data sources, active databases, signal databases, temporal databases, high-performance parallel data servers, and databases with real time properties. The laboratory has cooperative research projects for supporting engineering and telecom applications together with Swedish industry.

Research Organization and Laboratories

EIS - Economic Information Systems

Birger Rapp

The research areas of EIS involve, among other things, communication and transfer of information between people, as well as the development of suitable information systems for this purpose. The subject also deals with the use of modern information technology and the development of structures within organizations, together with the effects of information technology on people and organizations. This involves both questions concerning economic direction and control, and the capacity of people to take in and use information as well as training.

ESLAB - Embedded Systems Laboratory

Zebo Peng

ESLAB conducts research on the design and test of embedded systems, especially those consisting of tightly coupled hardware and software components. Special emphasis is placed upon the development of methods and tools for specification, modeling, synthesis, simulation, design for test, test synthesis, and hardware/software co-design. We are also concerned with the exploitation of systematic design methods and design automation techniques for industrial applications.

IISLAB - Laboratory for Intelligent Information Systems

Nahid Shahmehri

IISLAB conducts research in Intelligent Information Systems. The research topics include proposing principles, methods and tools for defining and constructing advanced information management systems tailored to present and future information technology. Current projects focus on information security, information retrieval and filtering, and representation, organization and processing of knowledge in distributed environments such as the World Wide Web.

KPLAB - Knowledge Processing Laboratory

Patrick Doherty

Research in KPLAB focuses on the theoretical and practical aspects related to the representation and processing of knowledge. Special emphasis is placed on the specification and implementation of deliberative/reactive architectures for autonomous artifacts. Current activities include the development of nonmonotonic temporal logics for reasoning about action and change, and the specification of higher level cognitive tasks such as planning and diagnosis. Special focus is placed on the study of unmanned aerial vehicle (UAV) architectures integrated with active vision systems. Other areas of interest include model-based simulation and real-time reasoning.

Research Organization and Laboratories

LIBLAB - Laboratory for Library and Information Science

Roland Hjerpe

Research at LIBLAB, is focused on long term studies of the interactions, positive and negative, between information technology and the generation, access to and use of documents and document collections. Within this very broad area the main objects of study are the issues of designing and using tools for access to large collections of documents, e.g. catalogues, search engines, systems for knowledge/document organization. The foci are on methods for description, organization and retrieval of documents and the content of documents, and on information/experience related behaviour.

LOGPRO - Logic Programming Laboratory

Jan Maluszynski and Ulf Nilsson

The long-term goal of LOGPRO is to improve the efficiency and ease of use of declarative languages; in particular languages involving logic and constraints. The group is currently involved in three research projects: (1) modeling and verification of dynamic systems by merging technology from the fields of model checking and constraint logic programming; (2) verification and declarative diagnosis of constraint logic programs and (3) models and algorithms for fault isolation in distributed and object-oriented dynamic systems.

MDA - People, Computers and Work

Toomas Timpka

The MDA group develops and studies information systems in working-life contexts, with a focus on applications in service organizations. The research has an interdisciplinary character and integrates methods from computer science, psychology and sociology. Specific areas of interest include computer-supported cooperative work, inter-organizational networks, economic evaluations of information systems and participatory design.

NLPLAB - Natural Language Processing Laboratory

Lars Ahrenberg

NLPLAB studies linguistic processing and knowledge representation from linguistic, computational and behavioral perspectives. Current applied projects concern spoken and multimodal natural-language dialogue systems and computer-aided translation.

PELAB - Programming Environments Laboratory

Peter Fritzson

PELAB is concerned with research in programming systems and software engineering, i.e. tools and methods for the specification, development and maintenance of computer programs. Some examples are: programming languages, debuggers, incremental programming environments and compilers, compiler generators, tools for debugging and maintenance of distributed and real-time systems, compilers and programming environments for parallel computers, high-level environments and mathematical modeling languages and systems for simulation and scientific

Research Organization and Laboratories

computing, program transformation systems, visualization and 3D animations of simulations, etc. Our view of programming environment research is rather pragmatic. We are primarily interested in developing and investigating new methods and tools that have potential for practical applications, e.g. in support systems for software specialists.

RTSLAB - Laboratory for Real-Time Systems

Anders Törne

The research in RTSLAB comprises tools, methods and architectures for design of software intensive real-time systems. This includes in particular timing analysis and synthesis tools based on discrete modeling methods, database modeling and languages for control and supervisory functions, multi-level architectures, high-level specification languages and tools for systems engineering. The laboratory also conducts application oriented research and case studies in cooperation with industry, for example in the areas of automation and process control.

TASLAB - Laboratory for Autonomous Systems

Dimiter Driankov

The research in TASLAB is aimed at developing the theoretical basis for the design and analysis of systems with high degree of autonomy. Enhancing the autonomy of unmanned vehicles and large industrial process control systems is of major interest. We address this problem in the context of a generalized three-layered system architecture and focus on topics such as the design and analysis of hybrid systems, discrete event control systems, and fuzzy control systems; fault identification and re-planning in sequential control, and software aspects for layered architecture autonomous systems.

TOSCA - Laboratory for Temporal-System Correctness and Algorithmics

Christer Bäckström

The research within TOSCA focuses mainly on theoretical aspects of temporal systems of various types, with two main sub-areas. Within the first of these we study the computational complexity of and design efficient algorithms for problems in action planning and temporal constraint networks as well as applications of these within automatic control and diagnosis. In the second sub-area we study modeling and verification of embedded systems. We use hybrid mathematical models (mixed continuous and discrete), as well as discrete (synchronous) languages for modeling purposes.

VITS - Development of Information Systems and Work Contexts

Stefan Cronholm/Göran Goldkuhl

VITS is a research group studying information systems development in an organizational context. Examples of research areas are: Business and communicative action theory, business process development, change analysis, information requirements analysis, evaluation of information systems and business activities, modeling methods, meta modeling, CASE and method supporting tools, information systems architecture, interorganizational information systems.

Faculty engaged in the graduate study programme



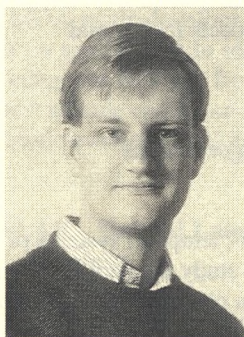
Leif Appelgren, Lic.Eng. Consulting professor (*adjungerad professor*), economic information systems. Previous academic appointments with Linköping and Handelshöyskolen, BI, Norway. Several previous affiliations. Current affiliation: Leif Appelgren Consulting AB.

Measurement and control related to economic crime prevention



Lars Ahrenberg, Ph. D., Uppsala 1987. Professor of computational linguistics. Group leader, NLPLAB. Research Committee chairman.

Syntax, semantics and pragmatics of natural language; natural language understanding; natural language interfaces; machine-aided translation.



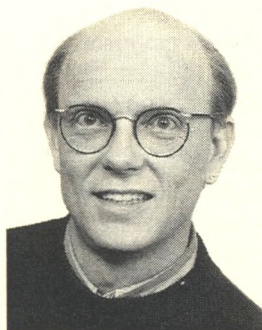
Christer Bäckström, Ph. D., Linköping 1992. Associate professor (*docent, universitetslektor*). Group leader, TOSCA.

Planning and temporal reasoning, algorithms and complexity, model-based diagnosis.



David Carr, Ph.D., Maryland 1995. Assistant professor (*universitetslektor*), human-computer interaction. Previous affiliations Maryland and Luleå.

User interface design, computer-supported cooperative work and groupware for teamwork.



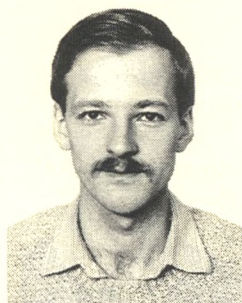
Nils Dahlbäck, Ph. D., Linköping 1992. Assistant professor (*universitetslektor*), cognitive science.

Natural language processing, especially empirically based computational models of discourse. Cognitive aspects of discourse coherence in man and machine. Intelligent user interfaces. Individual differences in cognitive abilities and their relationship to hypermedia navigation.



Patrick Doherty, Ph. D., Linköping 1991. Associate professor (*docent, universitetslektor*), logic and theoretical computer science. Group leader, KPLAB.

Artificial Intelligence, Knowledge Representation, Deliberative/Reactive Systems, Temporal Reasoning.



Włodzimierz Drabent, Ph. D., Warsaw 1985. Associate professor (*docent, universitetslektor*), computer science. A position also at the Institute of Computer Science, Polish Academy of Sciences.

Logic programming: negation, semantics, proving properties of programs, declarative diagnosis; programming languages semantics.



Dimiter Driankov, Ph. D., Linköping 1989. Associate professor (*docent, universitetslektor*), logic and AI. Group leader, TASLAB.

Reasoning under uncertainty, many-valued logics, approximate reasoning, fuzzy control & systems, autonomous agents.



Petru Eles, Ph. D., Bucuresti 1993. Assistant professor (*universitetslektor*), computer architectures. Previous affiliation: Technical University Timisoara, Romania.

Design of embedded systems, design automation for digital systems, real-time systems, hardware/software co-design, computer architectures, concurrent programming, hardware description languages and languages for system specification.



Henrik Eriksson, Ph. D., Linköping 1991. Associate professor (*docent, universitetslektor*), computer science. Previous affiliations: Stanford University, Stanford, CA, 1991-94. Part time at the Swedish Institute of Computer Science (SICS) 1996-97.

Knowledge-based systems, knowledge acquisition, medical informatics, software development environments, software reuse, Internet-based applications, Java programming.



Johan Fagerström, Ph. D., Linköping 1988. Assistant professor (*universitetslektor*), computer science.

Distributed systems, object-oriented programming, object-oriented analysis and design, operating systems.



Thomas Falk, Ph. D., Stockholm School of Economics 1976. Consulting Professor (*adjungerad professor*), economics of information technology. Several previous affiliations. Current affiliation: Federation of Swedish Industries.

Management of information technology.



Per-Olof Fjällström, Ph. D., Stockholm 1985. Associate professor (*docent, universitetslektor*), theoretical computer science. Group leader, ACTLAB. Previous affiliations: KTH and IBM.

Computational geometry with applications in computer-aided engineering and geoinformatics, design and analysis of sequential and parallel algorithms.



Dag Fritzson, Ph.D., Göteborg 1988. Consulting professor (*adjungerad professor*), engineering information systems especially scientific computing. Previous affiliation Chalmers University of Technology, Göteborg. Current affiliation SKF AB.

Modelling techniques, parallel simulation, visualization, machine element modelling, e.g. rolling bearings.



Peter Fritzson, Ph. D., Linköping 1984. Professor of programming systems and software engineering. Group leader, PELAB. Previous affiliations: Sun Micro Systems, USA, 1985-86, Linköping Univ. Hospital 1976-79. LiU Physics dept., 1975-76.

Programming environments and languages, design languages and simulation environments, scientific computing, debugging tools, incremental compilation technology, compiler generation, compilers and development tools for parallel hardware.



Anders Haraldsson, Ph. D., Linköping 1977. Associate professor (*bitr professor*), computer science. Head of the department of computer science. Previous affiliation: Uppsala.

Programming languages and systems, programming methodology, program manipulation, partial evaluation.



Mary Helander, Ph. D., SUNY Buffalo 1992. Acting professor of software engineering. Associate professor (forskarassistent), IKP Kvalitetsteknik. Group leader, ASELAB. Previous affiliations: Northeastern University, Boston, IBM Corporation.

Software reliability, software quality, network reliability, network location, operations research.



Roland Hjerpe, Director of Libraries, Mid Sweden University. Previous group leader, LIBLAB.

Library science and systems, hypertext and -media, knowledge organization and information retrieval, citation analysis and bibliometrics, personal and everyday life information/document management.



Sture Hägglund, Ph. D., Linköping 1980. Professor of knowledge-based systems. Deputy head of the department of computer science. Group leader, ASLAB. Previous affiliation: Uppsala.

Expert systems and artificial intelligence applications, database technology, human-computer interaction, intelligent tutoring systems and software engineering.



Olof Johansson, Ph. D., Linköping 1996. Assistant professor (forskarassistent), engineering information systems.

Engineering databases, complex product models.



Peter Jonsson, Ph.D., Linköping 1996. Associate professor (*docent, universitetslektor*), computer science.

Construction and analysis of algorithms. Complexity theory.

Erland Jungert, PhD Linköping 1980, consulting professor (*docent*) geographical information systems and computer science. Main affiliation: The Defense Research Establishment (Foa).

Geographical information systems, database technology, visual languages.



Arne Jönsson, Ph. D., Linköping 1993. Assistant professor (*universitetslektor*), computer science. Director of undergraduate studies for the Cognitive Science program. Previous affiliation: Monash University, Clayton, Australia, 1994-95.

Artificial intelligence, natural language processing, especially empirically based computational dialogue models.



Mariam Kamkar, Ph. D., Linköping 1993. Acting professor of software engineering. Deputy head of the department of computer science.

Software maintenance, software testing and analysis, program debugging, program analysis, optimization in compilers, multiparadigm programming languages.



Krzysztof Kuchcinski, Ph. D., Gdansk 1984. Associate professor (*docent, universitetslektor*), computer systems. Group leader, CADLAB. Previous affiliation: Technical University of Gdansk, Poland.

Computer architecture, computer-aided design of digital systems, VLSI, design for testability.



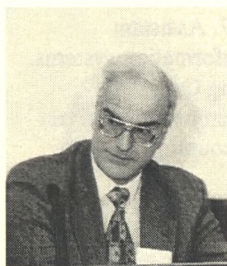
Patrick Lambrix, Ph.D., Linköping 1996. Assistant professor (*universitetslektor*).

Intelligent Information Systems: knowledge representation, organization and management of information, common-sense reasoning, knowledge-based information retrieval, information extraction.



Bengt Lennartsson, Ph. D., Göteborg 1974. Associate professor (*universitetslektor*), software engineering. Group leader, PELAB 1981-88 and head of the department of computer science 1983-90. Previous affiliation: Carlstedt Elektronik AB, Göteborg 1992-94. Current affiliation ITN, Campus Norrköping.

System development models, development of complex systems, organizational learning.



Bertil Lyberg, Ph.D., Stockholm 1981. Consulting professor (*adjungerad professor*), speech technology. Manager of spoken language processing, Telia Research AB. Research Affiliate, Research Laboratory of Electronics, M.I.T. 1982-83. Text-to-speech conversion, speech recognition, speech-to-speech translation, prosody in speech synthesis and recognition.



Jan Maluszynski, Ph. D., Warsaw 1973. Professor of programming theory. Several previous affiliations.

Logic programming, formal language theory, integration of programming paradigms.



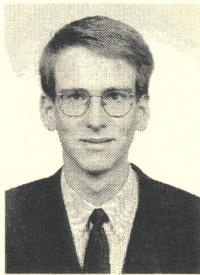
Anna Moberg, Ph.D., Linköping 1997. Assistant professor, (*vik. universitetslektor*), economic information systems.

IT and organizational design, communication, telework, non-territorial office.

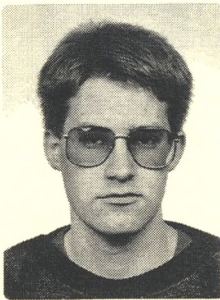


Simin Nadjm-Tehrani, Ph. D., Linköping 1994. Assistant professor (*universitetslektor*), computer science.

Modelling and formal verification of embedded systems, hybrid (discrete/continuous) models, rule-based and synchronous languages, temporal logic.



Fredrik Nilsson, Ph. D., Linköping 1997. Assistant professor (*forskarassistent*), economic information systems. Manager at Deloitte & Touche Consulting Group. Strategy and management control, environmentally-driven business development, valuation and accounting of intangible assets.



Henrik Nilsson, Ph. D., Linköping 1998. Assistant professor (*vik. forskarassistent*), computer science.

Functional programming languages, programming language implementation, functional programming, declarative debugging.



Ulf Nilsson, Ph. D., Linköping 1992. Associate professor (*docent, universitetslektor*), computer science. Deputy head of the department of computer science. Group leader, LOGPRO. Previous affiliation: State University of New York at Stony Brook, USA.

Logic programming and deductive databases; Model checking; Evaluation strategies for query processing; Program transformation and abstract interpretation.



Kjell Ohlsson, PhD, Umeå 1982. Part-time professor of human-computer interaction. Previous affiliations Umeå, Luleå and others.

Human-computer interaction, usability engineering, cognitive ergonomics, decision making, psycho acoustics.



Nils-Göran Olve, Econ Dr., Stockholm 1977. Consulting professor (*adjungerad professor*), management control. Positions with the Stockholm School of Economics and EIASM in Brussels. Partner in Cepro Management Consultants since 1986.

Management issues arising from IT-enabled business change, especially how accounting, control, and pricing could be modified to provide information and incentives appropriate for the new business situation.



Kjell Orsborn, Ph. D., Linköping 1996. Assistant professor (*forskarassistent*), engineering information systems. On leave to Intelligent Engineering Systems Laboratory (IESL), MIT, Cambridge, USA.

Database technology for scientific and engineering applications, specifically computational database technology. Extensible database technology applied to the fields computer-aided design, computational mechanics, and product data management.



Zebo Peng, Ph. D., Linköping 1987. Professor of computer systems. Group leader, ESLAB.

Design and test of embedded systems, electronic design automation, design for testability, hardware/software co-design, real-time systems, computer architecture, VLSI.



Birger Rapp, Econ. Dr., Stockholm 1974, Professor of economic information systems. Group leader, EIS. Among many other appointments president of the board of the Swedish Teleworking Association, Distansforum and program director in Management and Economic Information Systems at IMIT.

Accounting, business control, agency theory, IT and organization, production, economics.



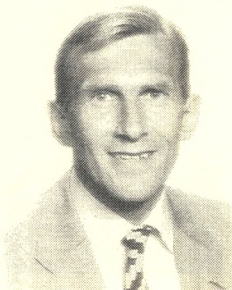
Tore Risch, Ph. D., Uppsala 1978. Professor of engineering databases. Group leader, EDSLAB. Previously at Uppsala University, IBM Almaden Research Lab. (San Jose, CA), Stanford Research Institute, Syntelligence Inc. (Sunnyvale, CA), HP Laboratories (Palo Alto, CA), and Stanford University.

Database support for engineering and scientific applications, e.g., object-oriented databases, heterogeneous databases, active databases, and real-time databases.

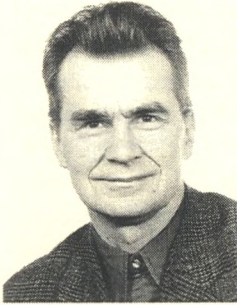


Nancy Reed, Ph.D., University of Minnesota 1995. Assistant professor (*forskarassistent*), computer science. Previously at University of California, Davis.

Diagnostic problem-solving methods, modelling human expert reasoning, currently involved in high level specification languages for software agents in interactive simulations.

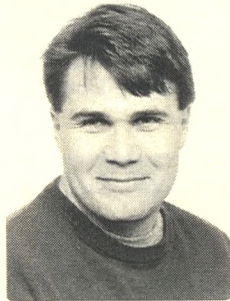


Rolf Rundfelt, Ph. D., Stockholm 1974. Consulting professor (*adjungerad professor*), economic information systems, especially in Swedish and international Financial Accounting. Docent in Business Administration, University of Stockholm. Since 1966, lecturer at the University of Stockholm.



Erik Sandewall, Ph. D., Uppsala 1969. Professor of computer science. Prorector of Linköping University. Several previous affiliations.

Representation of knowledge with logic, reasoning about action and change, cognitive robotics, autonomous agents.



Kristian Sandahl, Ph. D., Linköping 1992. Consulting Associate professor (*docent*), computer science. Main affiliation Ericsson Radio Systems AB.

Knowledge management, knowledge engineering, industrial software engineering, quality improvement paradigm, empirical research methods.



Nahid Shahmehri, Ph. D., Linköping 1991. Professor of computer science. Group leader, IISLAB. Director of graduate study programme. Previous affiliation: Carlstedt Elektronik AB, Göteborg.

Information management, information retrieval and filtering, information extraction, information security, workflow management, CSCW.



Åke Sivertun, Ph. D., Umeå 1993. Assistant professor (*forskarassistent*) at LIBLAB. Assistant professor (*universitetslektor*) at Högskolan i Kalmar.

Geographical Information Systems - GIS. Communication of complex data and linking multi disciplinary models in GIS. Research in environmental programs, programs for medical geography, physical planning and decision support.



Martin Sköld, Ph. D., Linköping 1997. Assistant professor (*forskarassistent*), engineering information systems. Engineering databases, active, temporal, and real-time database systems.

Telecommunication and Computer Integrated Manufacturing (CIM).



Lena Strömbäck, Ph.D., Linköping 1997. Assistant professor (*forskarassistent*), computational linguistics. Director of graduate studies for the HMI programme.

Natural language understanding, tools for grammar development, unification-based formalisms, models for natural language.



Toomas Timpka, MD., Stockholm 1983, Ph. D., Linköping 1989. Professor of information systems development. Group leader, MDA. Acting Professor of Social Medicine and Associate Professor of Medicinal Informatics.

Hypermedia, computers and society, human-computer interaction, systems development.



Anders Törne, Ph. D., Uppsala 1980. Associate professor (*universitetslektor*), computer support in automation. Group leader, RTSLAB. Previously at ABB Corporate Research, Västerås. CENIIT manager from 1992.

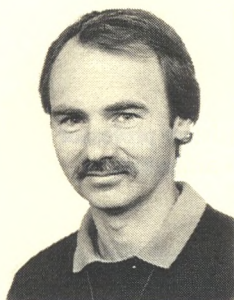
Tools, methods and architecture for real-time system design. Applications in automation and embedded systems. Real-time programming and specification languages. Robot programming.

Guest researchers and affiliated faculty engaged in the graduate study programme



Sten F. Andler, Ph. D., Carnegie-Mellon University 1979. Professor of computer science, Höskolan i Skövde, (docent LiU), distributed systems, real-time systems, operating systems. Previous affiliations: IBM Software Solutions (1992-93) and Almaden Research Center (1979-92), San Jose, CA.

Distributed real-time systems, real-time databases, active real-time databases, distributed databases, real-time operating systems.

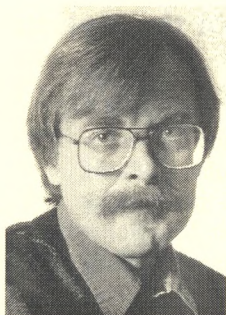


Göran Goldkuhl, Ph. D., Stockholm 1980. Professor of informatics, Jönköping International Business School. Research director VITS network. Research director at Centre for studies on Man, Technology and Organization. Previous affiliations: Stockholm, Göteborg.

Theories/methods on business process and information systems development, method modelling and renewal, business and communicative action, qualitative research methods.

Witold Litwin, Professor University Paris 9. Guest professor Linköping, Stanford, Berkeley, Santa Clara univ.

Distributed scalable data structures (SDDSs), multidatabase systems, storage structures, query languages.



Witold Lukaszewicz, Ph.D., Warsaw University 1979. Guest professor. On leave from the Institute of Informatics, Warsaw University, Poland.

Knowledge representation, non-monotonic reasoning, programming methodology.



Jonas Löwgren, Ph. D., Linköping 1991. Associate professor (*docent, universitetslektor*), human-computer interaction. On leave since 97/98.

Human-computer interaction, usability-oriented systems development, interaction design.



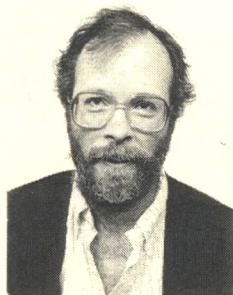
Jacek Malec, Ph.D., Wrocław 1987. Assistant professor (*forskarassistent*) in TASLAB. Previous affiliation: Technical Univ. of Wrocław, Poland.

Reactive systems, autonomous systems, system theory, knowledge representation., artificial intelligence.



Anders G. Nilsson, Econ Dr., Stockholm 1991. Professor of Informatics, University of Karlstad. Associate professor (docent, LiU) of information systems development in 1995. Acting professor of economic information systems in 1992. Previous affiliations: KTH, University of Stockholm and Stockholm School of Economics.

Application packages, business modelling, business process reengineering (BPR), information management, ISD methods, IS/IT strategies, maintenance management.



James M. Nyce, Ph.D., Brown 1987. Guest professor (docent, LiU) computer and information science. Associate professor, School for Library and Information Management, Emporia State, Emporia, KS, USA. Previous affiliation Brown.

Work and knowledge (medicine and academia); tradition, innovation and technology; hypertext and visual resource development paths.





LINKÖPINGS UNIVERSITET

Department of Computer Information Science (IDA)
S-581 83 Linköping
Sweden