Fall

DEPARTMENT OF COMPUTER AND INFORMATION SCIENCE (IDA)

INTERNATIONAL GRADUATE SCHOOL IN COMPUTER SCIENCE

COURSES 97/98

Fall

LINKÖPINGS UNIVERSITET, SWEDEN

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**
- **Library and Information Science**
- **Computational Linguistics**
- **Economic Information Systems**
- **Information Systems Development**
- **Engineering Information Systems**

The Department also participates in the new graduate schools Excellence Center in Computer and Systems Engineering (ECSEL) 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 17 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 abour 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

Administrator of Graduate Studies

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

Graduate Courses Fall 1997

Course	Course literature	Author	Aug	Sept	Oct	Nov	Dec	Day	Time
Affärsprocessanalys Göran Goldkuhl, 5p	Articles			Starts Sept 9			Ends Dec 9	Separate schedule Location: Jönköping	
Aspekter av vetenskapligt skrivande Ulf Nilsson, 3p	1. Från Tanke till Text 2. Academic Writing for Graduate Students	1. Jarric and Josephson 2. Swales and Feak		Starts Sept 24	Oct 1 Oct 8 Oct 22 Oct 29			Wed	10-12
Business Modeling i teori och praktik Anders G Nilsson, 5p	Business in the Information Age - Heading for New Processes	H. Österle	Starts Aug 28				Ends Dec 16	Aug 28 Oct 8 Oct 9 Dec 16	13-17 10-17 13-17 10-17
Data Mining and Knowledge Discovery Ankica Babic, 3p	Selected articles			Starts Sep 29				Mon	13-16
Discrete Systems Modeling Krzystof Kuchcinski, 4p	1. Petri Net Theory and the Modeling of Systems 2. Introduction to automata theory, languages, and computation 3. Parallel Program Design - A Foundation 4. The Temporal Logic of Reactive and Concurrent Systems	1.J. L. Peterson 2. John E. Hopcroft , Jeffrey D. Ullman 3. K.M. Chandy, J. Misra 4. Z. Manna, A. Pnueli			Starts Oct 6	Ends Nov 21		Mon Fri	10-12 10-12
Informationsteknik och Management Thomas Falk, 5	Articles				Starts Oct 21		Ends Dec 17	Oct 21 Oct 22 Oct 28 Oct 29 Dec 17	13-19 8-12 13-19 8-12 10-17
Introduction to Research Methodology in Computer Science Sture Hägglund, 2p	1. What is this thing called science? 2. Survival strategies for young scientists	1. Chalmers 2. Sindermann			Starts Oct 2	Ends Nov 20		Thur	10-12
Language Engineering Systems Lars Ahrenberg, - 5p	1. Evaluating Natural Language Processing Systems. An Analysis and Review.Lecture Notes in Artificial Intelligence 1083 2. Survey of the State of the Art in Human Language Technology Ch. 2-4, 6-13	Laren Sparck Jones & Julia R. Galliers Ron Cole et al		Starts Sept 8	Ends Oct 30			Mon w.37-40 Thur w.37-44	10-12 10-12
Modeling and Analysis of Real Time Systems Anders Törne, 5p	Articles and reports					Starts Nov 97, Ends March 98		4 days in Linköping, 4 days in Uppsala, the dates are to be decided	
Object Oriented Modeling Languages for Dynamic Systems Peter Fritzson, Johan Gunnarsson, 3p	Articles and book draft on Modelica					Starts Nov 3	Ends Dec 22	Mon	13-16
Real Time Systems Engineering Anders Törne, 5p	Selected papers				Starts Oct 1		Ends Dec 17	Fri	13-15
Selected Notions in the Theory of Computing Jan Maluszynski, 6p	1. The science of computing: exploring the nature and power of algorithms 2. Petri net theory and the modeling of systems	1. David Hazel 2. James Lyle Peterson			Starts Oct 7		Ends Dec 18	w.41 Tues w.43-51 Tues, Thur	 10-12 10-12 10-12

Contents

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.

The department also participates in three special graduate schools aiming for interdisciplinary studies preparting also for a career outside the university. HMI, Human Machine Interaction, starting 1997 in cooperation with divisions of Industrial Ergonomics and Communication studies. ECSEL, Excellence Center in Computer Science and Systems Engineering, started 1996 in cooperation with primarily the Department of Electrical Engineering. IMIE, International Graduate School of Management and Industrial Engineering, has been in operation a few 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 of 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 (forskarstudierektor). However, most of the administration and organization rests upon the director of research administration (Lillemor Wallgren). Most Ph. D. students are employed by the department full time. They assist in undergraduate courses and other internal assignments of the laboratories, up to about 15 - 30% of their time. The rest of the time is spent on courses and thesis project.

Graduate Programme

in Computer and Information Science 1997/1998

This program contains the following types of courses:

- **Graduate Courses at the International Graduate School of Computer and Information Science**
- **Recommended Master Courses**
- **Laboratory-Oriented 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, etc. Those courses will be separately announced.

The following acitvities 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.

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 24, 1997

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

Affärsprocessanalys

Recommended for: **Lectures:** 33 h
Kursen vänder sig till informatik-doktorander vid IHH och informationssystemutvecklings-doktorander från IDA. Kursen är också öppen får andra doktorander vid IHH och IDA samt från CMTO, Linköpings universitet.

The course last ran:
New course.

Goals:
Kursen syftar till att öka förståelsen för affärsprocesstänkande och hur detta kan utgöra en grund för verksamhets-/systemutveckling och metoder för detta.

Prerequisites:
Önskvärt (men inte nödvändigt) med genomgången kurs i förändringsanalys.

Organization:
Föreläsningar, seminarier. Kursen kommer huvudsakligen att äga rum på IHH, Jönköping.

Contents:
Affärsprocesstänkande som grund för verksamhets-/systemutveckling

Affärsprocessinriktad förändringsanalys; metoder för affärsprocessinriktad verksamhetsutveckling

Affärsprocessbegreppet; transformation vs uppdrag

IT-stöd för affärsprocesser och affärskommunation

Kvalitet i affärskommunikation

Affärsaktsteori; koppling till teorier om kommunikativt handlande och affärsrelationer

Affärsgeneriska faser, affärsgeneriska begrepp

Literature:
Artiklar från och med anknytning till egen pågående forskning.

Teachers:
Göran Goldkuhl, Mikael Lind, Owen Eriksson, Marie-Therese Lundmark.

Examiner:
Göran Goldkuhl.

Schedule:
Kursen genomföres under sep-dec 1997.

Tisdag 9/9 kl 10.00-12.00 samt kl 13.00-15.00

Tisdag 16/9 kl 13.00-17.00

Onsdag 17/9 kl 09.00-13.00

Onsdag 1/10 kl 13.00-17.00

Torsdag 2/10 kl 09.00-13.00

Onsdag 15/10 kl 10.00-12.00 samt 13.00-15.00

Tisdag 28/10 kl 10.00-12.00 samt 13.00-15.00

Tisdag 9/12 kl 10.00-12.00 samt 13.00-16.00 (Examinationsseminarium)

All undervisning genomförs på IHH, Jönköping. Lokaler meddelas senare.

Examination:
Olika typer av examinationsuppgifter kommer att kunna väljas: 1) empiriskt inriktad eller

2) teoretiskt inriktad.

Credit:
5 p.

Arrangörer: Kursen samarrangeras av Internationella handelshögskolan (IHH) i Jönköping och Institutionen för datavetenskap (IDA), Linköpings universitet.

Aspekter av vetenskapligt skrivande

Recommended for:
Alla doktorander inom området datavetenskap.

Lectures: 10 h

The course last ran:
1996/97.

Goals:
Utgöra basen för skrivande av vetenskapliga artiklar och avhandlingar. Att öka medvetandet om problem (och möjligheter) i den skrivande processen. Att ge förslag till förbättringar av stilen i skrivandet, uppläggningen av en artikel och hur man kan dra fördel av typografi för att framföra innehållet.

Prerequisites:
Inga.

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 titelns, 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 sambandsignaler. 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:
Jarrick och Josephson, Från Tanke till Text, Studentlitteratur

Swales and Feak, Academic Writing for Graduate Students, Michigan Press

Examiner:
Ulf Nilsson.

Schedule:
Sept-Okt.

Examination:
Aktiv närvaro samt inlämningsuppgifter.

Credit:
3 points.

Note:

Kursen ges på svenska.

Business Modelling i teori och praktik

Recommended for:
Doktorander och forskarstuderande vid

- Ekonomiska informationssystem (EIS)
- Informationssystemutveckling (VITS/ISU)
- IMIE forskarskola

The course last ran:
New course.

Goals:
Öka förståelsen för modellanvändning vid förändringsarbete inom företag och organisationer med speciell fokus på sambandet mellan

- affärsutveckling
- verksamhetsutveckling
- systemutveckling

Prerequisites:
Grundkunskaper i företagsekonomi eller systemutveckling

Organization:
Kursen består av koncentrerade föreläsningar, litteraturbevakningar, teoretiskt modelleringsarbete och praktikfallsanalyser.

Contents:
Följande teman kommer bl.a. att beröras:

- Modellers och metoders roll vid förändringsarbete
- Olika perspektiv och aspekter vid modelleringsarbete
- Teorier från företagsekonomi och informationssystem för modellering
- Metodkombinationer (t.ex. metodkedjor och metodallianser)
- Kvalitetssäkring av modeller och metoder
- Modelleringsarbete inom mindre och större företag
- Praktikfall inom Business Modelling

Examiner:
Anders G. Nilsson

Schedule:
Specifika datum under hösten 1997, se schema.

Examination:
Skriftliga rapporter och uppsatser. Seminarier.

Credit:
5 points.

Data Mining and Knowledge Discovery

Recommended for:
It applies to other graduate schools, as well.

Lectures: 10 h

The course last ran:
New course.

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:
Selected articles.

Examiner:
Ankica Babic

Schedule:
In the fall. Start in September.

Examination:
Active participation, seminar presentations.

Credit:
3 points.

Discrete Systems Modeling

Recommended for:
PhD students.

Lectures: 30 h

Approproate for ECSEL as Specialized Study Course.

The course last ran:
New course.

Goals:
The course will present basic methods used to model and analyze discrete systems.

Prerequisites:
Discrete mathematics, basic course in logic.

Organization:
Seminars.

Contents:

- Introduction- 4h (Jacek Malec)

- Introduction to system modeling
- Classification to static and dynamic systems (discrete state, discrete time

- Stationary systems

- Causal systems

- Goal-searching systems

- Stability, Observability and Controllability

- Analysis, synthesis and verification of discrete systems

- Transition systems- 8h (Simin Nadjm-Tehrani)

- Petri nets- 8h (Zebo Peng, Kris Kuchcinski)

- Automata- 8h (Inger Klein)

- Temporal logic- 8h (Jacek Malec)

Literature:
J. L. Peterson: "Petri Net Theory and the Modeling of Systems", Prentice-Hall, 1981.

John E. Hopcroft and Jeffrey D. Ullman: "Introduction to automata theory, languages, and computation", Addison-Wesley, 1979.

K. M. Chandy and J. Misra: "Parallel Program Design - A Foundation", Addison-Wesley, 1988.

Z. Manna and A. Pnueli: "The Temporal Logic of Reactive and Concurrent Systems", Springer Verlag, 1991.

Selected papers.

Examiner:
Krzysztof Kuchcinski.

Schedule:
October-November 1997, 4 hours/week.

Examination:
Obligatory homework.

Credit:
4 points.

Informationsteknik och Management

Recommended for:
Forskarstuderande inom IDA, IMIE och EKI.

Lectures: 36 h

The course last ran:
1996.

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 1997. Den består av föreläsningar, litteraturstudier, 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 företagens konkurrensförmåga och för en effektiv offentlig förvaltning. IT är en strategisk resurs och information ett viktigt konkurrensmedel. I alltstörre utsträckning ingår IT i nya affärsidéer, 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 arbets -sätt och affärsprocesser.

Examiner:
Thomas Falk

Schedule:
Oktober-december 1997 (21/10, 22/10, 28/10, 29/10, 17/12).

Examination:
Skriftlig tentamen, skriftlig bokrecension, godkänd uppsats samt opposition.

Credit:
5 poäng.

Introduction to Research Methodology in Computer Science

Recommended for:
New graduate students.

Lectures: 16 h

The course last ran:
Fall 1996.

Goals:
To give an introduction to the philosophy of science, the special characteristics of computer science research and to discuss aspects of graduate studies and scientific activities.

Prerequisites:
None.

Organization:
Lectures and seminars.

Contents:
Introduction to the philosophy of science. Scientific writing, publication and information retrieval. Science and technology, methodological issues. Survival strategies for young scientists.

Literature:
Chalmers: What is this thing called science?

Sindermann: Survival strategies for young scientists.

Lecture Notes.

Teachers:
Sture Hägglund et al.

Examiner:
Sture Hägglund.

Schedule:
October - November 1997.

Examination:
Written examination and seminar activity.

Credit:
2 points.

Language Engineering Systems

Recommended for:
Anyone interested.

Lectures: 12 h

The course last ran
New course.

Goals:
The course will provide a thorough overview of state-of-the-art in language technology and hands-on experience with advanced LE systems.

Prerequisites:
A basic course in Computational linguistics (such as TDDA86).

Organization:
Lectures and practical exercises.

Contents:
Language engineering techniques and methodologies; evaluation methodology;

Usability aspects;

Application areas: reading aids, translation aids, writing aids, information extraction, dialogue systems;

Generic tools and resources (tagging systems, lexical databases, grammars, core engines, ...)

Literature:
Ron Cole et al (eds.). Survey of the State of the Art in Human Language Technology, chapter 2-4, 6-13. (http://www.hgw.tu-berlin.de./~neigel/SpeedTech/).

Artiklar.

Teachers:
Lars Ahrenberg, Lena Strömbäck.

Examiner:
Lars Ahrenberg.

Schedule
Autum -97, weeks 37-44. Weeks 37-40 Monday 10.00-12.00 and weeks 37-44 Thursday 10.00-12.00.

Examination:
A project + exercises.

Credits:
Up to 5 points.

Note:
The course is also offered to 4th year students of the Cognitive Science program. (Master's program in Språkinformatik).

Modeling and Analysis of Real-Time Systems

Recommended for: **Lectures:** 48 h
Graduate students in computer science and computer systems. Appropriate for ECSEL and ARTES related graduate students.

The course last ran:
New course.

Goals:
The goal is to make the graduate student aware of and understand the differences between traditional software engineering and real-time systems engineering. The student shall also become more familiar with one of these by applying it to a realistic scenario. These projects are performed in groups of suitabe size.

Prerequisites:
Basic course in real-time systems or similar

Basic course in automata theory and logic

Organization:
The course will be given as 20 lectures divided into two parts. The course is jointly organized by DoCS, UU and IDA, LiU. Half of the course will be given in Uppsala and half in Linköping in 8 full day (6 hr per day) lectures and 2 seminar days for practical exercises.

Contents:
The course contains two parts.

The first part is about execution time analysis. All types of real-time modeling and analysis requires execution time analysis to provide the necessary timing information. Methods and tools for prediction of program execution times will be reviewed. Simple systems, as well as more advanced systems including memory hierarchies and pipelines will be considered. In addition, the course will also address scheduling theory, in particular, static-cyclic scheduling, fixed-priority scheduling (FPS), and dynamic (e.g. earliest-deadline-first) scheduling, as well as earlier and non-real-time scheduling methods. Particular emphasis is given to the response time analysis used in FPS. There will be a project applying FPS in the development of a simple real-time system.

The second part is about formal modelling and analysis of real-time systems, in particular specification and automatic verification techniques for safety-critical systems. The goal of this part is to give an overview on recent advances in this area. The central topics include: finite-state models of real-time systems, such as timed automata, timed Petri Nets, hybrid automata and timed process algebras, various analysis methods and algorithms for these models and case studies. We will concentrate on the theory of timed automata and model-checking techniques and tools for timed automata. There will be a project work, based on the automatic verification tool UPPAAL developed at the department of computer systems in Uppsala University.

Literature:
Articles and reports.

Teachers:
Hans Hansson, Wang Yi (Uppsala universitet).

Examiner:
Anders Törne.

Schedule:
November 1997 to March 1998. 4 full days in Linköping and 4 full days in Uppsala. Dates are to be decided.

Examination:
Project.

Credit:
5p (3 litterature + 2 practical).

Note:

This course have been suggested from Uppsala to be classified as an ARTES course for 97/98. The suggestion to make it a joint course is preliminary until confirmed by the course leaders from UU.

Anders Törne is the responsible person from IDA.

Object Oriented Modeling Languages for Dynamic systems

Recommended for:
Appropriate for ECSEL, IDA students, ISY students.

Lectures: 24 h

The course last ran:
New course.

Goals:

To give an overview of modern equation-based object oriented modeling languages, with emphasis on the new language Modelica, and how to model complex dynamic systems.

Prerequisites:
General programming and engineering background.

Organization:
Lectures and exercises/mini-project.

Contents:
What is an Object Oriented Modeling language?

The concepts of model, simulation, simulation experiment

Different forms of ordinary differential equation systems

Object model. Connection structure. Units. Type system and type checking

Connection of subsystems

Integration of discrete and continuous system modeling

Examples of realistic application models, e.g. robots, airplanes, etc

Compilation techniques for modeling languages

Literature:
Articles and book draft on Modelica.

Teachers:
Peter Fritzson, Johan Gunnarsson.

Examiner:
Peter Fritzson.

Schedule:
Nov-Dec 1997.

Examination:
Exercises and mini project.

Credit:
3 points.

Real Time Systems Engineering

Recommended for: **Lectures:** 24 h
Graduate students in computer science and computer systems.

Appropriate for ECSEL and ARTES related graduate students.

The course last ran:
New course.

Goals:
The goal is to make the graduate student aware of and understand the differences between traditional software engineering and real-time systems engineering. The student shall also become more familiar with one of these by applying it to a realistic scenario. These projects are performed in groups of suitable size.

Prerequisites:
Basic knowledge in software engineering - structured analysis and design, object-oriented design

Basic knowledge in real time/process programming

Organization:
The course will be given as 12 lectures on different approaches.

10 lectures will present different methods and tools

2 lectures will compare and analyse the differences.

The students will perform one practical exercise using one of the methods or tools presented.

Contents:
8 approaches to real-time system design and analysis will be presented:

- Systems Software Engineering Methodology (SSEM)
- Real Time Structured Analysis and Design (RTSAD)
- Design Approach for Real-Time Systems (DARTS)
- Object Oriented Design
- HRT-HOOD
- IEEE 1220
- Raphsody - UML
- O4S

Literature:
Selected papers.

Examiner:
Anders Törne.

Schedule:
October to December 1997.

Examination:
Home exam and project documentation.

Credits:
5p (3 literature + 2 practical).

Selected Notions in the Theory of Computing

Recommended for:
Foundational course for ECSEL.

Recommended for PhD students with a non-computer-science background, e.g. students from IMIE.

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

Organization:
The course will include lectures and problem solving 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. The notion of algorithm. We discuss formalisations of the notion of algorithm, in particular Turing machine. We show its relevance for studying decidability and tractability of problems. We present examples of intractable problems, among others the Constraint Satisfaction Problem, which are very relevant in many applications.

4. Analysis of algorithms We discuss complexity of some comonly used algorithms on data structures. We also address the question how to deal with untractable problems.

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.

Examiner:
Jan Maluszynski.

Schedule:
Oct - approx Dec 97.

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

Cognitive Modeling

Recommended for: **Lectures:** 30 h
Students in computer science and cognitive science.

The course last ran:

New course.

Goals:

To provide insight into various modeling paradigms, in particular their scope and limitations, and give hands-on experience with currently available tools and progamming environments used for modeling.

Prerequisites:
Introductory course in AI. Familiarity with basic AI programming techniques.

Organization:
Lectures introducing basic modeling concepts and theoretical problems

Seminars on how different modeling paradigms relate to each other

Small modeling assignments using available modeling tools

Contents:

Systems analysis at different processing levels; should we model overall behavior or its underlying cognitive mechanisms? Virtues and disadvantages of connectionistic versus symbolic models. Computational models, dynamic simulations of processes in cascade. Unified cognitive architectures, comparison between ACT-R and SOAR. Cognitive neuroscience as an approach to modeling; assumptions about modularity and additivity in neuropsychology.

Literature:
Selected articles and book chapters.

Teacher:
Rita Kovordanyi.

Examiner:

Nils Dahlbäck, Rita Kovordanyi, invited lecturers

Schedule:

Spring 1998

Examination:
Written essay on a couple of self-selected exam questions or a short term paper in which an existing model is critically scrutinized (5 p).

Completion and written presentation of a small modeling project. (2 p)

Credit:
5 + 2 points.

Functional Programming

Recommended for:
Graduate students in Computer Science.

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 lazyfunctional 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:
Part I: Basic Functional Programming

Undergraduate programming course(s).

Undergraduate data structure and algorithm course such as TDDB57.

Part II: Advanced Functional Programming

Either part I of this course, or as for part I plus TDDB92/93 Programming in Incremental Systems or similar.

Part II: Advanced Functional Programming

Either part I of this course, or as for part I plus TDDB92/93 Programming in Incremental Systems or similar.

Organization:
Two parts: Basic Functional Programming and Advanced Functional Programming.

Part I is intended as an introductory functional programming course for people who do have the required programming background, but who do not have any previous experience of functional languages or languages with a functional flavour such as Lisp or Scheme. Students with previous functional experience (e.g. students from the C or D programmes) should only expect credits for part II of the course.

Part I: Lectures, assignments.

Part II: Lectures, assignments, presentations, project.

Contents:
Part I:

* Fundamental concepts

* Recursion

* Higher order functions

* Lists

Part II:

* 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
* Case studies
* Compulsory project

Literature:
To be decided.

Teachers
Henrik Nilsson

Schedule:
Early spring 1998.

Examination:
Part I: Compulsory assignments.

Part II:

* 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:
Part I: 1 point. Part II: 5 to 7 points depending on the size of the project.

Note:
Please state whether or not you need to take part I of the course.

Hardware/Software Co-Design

Recommended for: **Lectures:** 24 h
Ph.D. students in computer science and computer systems ECSEL students in the SCORE and STEM areas.

The course last ran:
New course.

Goals:
To present techniques for the concurrent design, or co-design, of hardware and software. Special emphasis will be placed upon methods and tools used for the development of embedded systems which are dedicated to specific applications and consist of tightly coupled hardware and software components.

Prerequisites:
Basic knowledge of software development and digital hardware.

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

Contents:
Design of embedded systems

Hardware/software co-specification

System modeling

Architecture selection

Application-specific instruction set processors

Embedded signal processing systems

Embedded control systems

Hardware/software partitioning

Co-synthesis and code-generation techniques

Interface analysis and synthesis

Co-simulation and co-verification

Emulation and prototyping

Survey of co-design research

Case studies.

Literature:
To be decided.

Teachers:
Zebo Peng and invited lecturers.

Examiner:
Zebo Peng.

Schedule:
March - May 1998.

Examination:
Term paper and seminar presentation.

Credit:
4 points.

Industrial Project Management

Recommended for:

Alla.

Lectures: 6 heldagar

The course last ran:

1996/97.

Goals:
Kursen ges av IMIE graduate school.

Organization:
Deltagarantalet är begränsat till ca 18 personer 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.

Examiner:
Christian Berggren, tf professor industrial organization.

Schedule:
Preliminärt vecka 16 till 22.

Examination:
Muntlig examination.

Credit:
5 poäng.

Intelligent Software Agents

Recommended for:
Graduate Students. (Orienteringskurs C4)

Lectures: 30 h

The course last run:
New course.

Goals:

Intelligent software agent technology has emerged as a new paradigm for software design and development with a wide range of application areas, such as engineering software and intelligent information systems. At the conclusion of the course the participants will have a good understanding of the principles and use of intelligent software agents technology and have gained experience in the design and implementation of intelligent agents in various applications.

Prerequisites:
General knowledge of programming languages, artificial intelligence, software design and architecture.

Organization:
Part I: A combination of lectures + seminars (prepared by the participants) + practical exercises.

Part II: Project.

Contents:
Part I: Topics in intelligent software agents

- introduction to intelligent agents and agent-based technology/systems
- properties and capabilities of agents
- design and architecture of agent-based systems
- multi-agent systems
- agent implementation languages
- agent communication and agent-communication languages
- agents in complex environments
- applications areas (intelligent information systems, engineering systems, etc.)

Part II: Project

- design, implementation, and evaluation of a multiLiterature:

Teachers:
Nahid Shahmehri and Patrick Lambrix.

Examiner:
Nahid Shahmehri (part I) and Patrick Lambrix (part II).

Schedule:
Start in Feb. 98.

Examination:
Part I: seminar presentations, active participation, exercises, and a term paper. Part II: project (re-quirement and design specifications, implementation, and experience report).

Credit:
Part I: 4 and Part II: 3+1. Part II is optional.

Note:
Level (for undergraduate students): D-level = Master's level.

IT-ekonomi och informationsekonomi

Recommended for: **Lectures:** 34 h
EIS. Lämplig för andra intresserade med grunder i företagsekonomi enligt ovan.

The course last ran:
1996.

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 ämnen såsom: IT-investeringskalkylering i en kommun; Hur fyra företag värderat nyttan vid EDI-införande; IT in IKEA; Värdera IT-nyttta vid distansarbete. 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; Hogbin & Thomas, Investing in Information Technology.

Teachers:
Nils-Göran Olve plus gästföreläsare.

Examiner:
Nils-Göran Olve.

Schedule:
Period 4. Våren 1998.

Examination:
Godkända arbetspapper, framlagda vid seminariet.

Credit:
3 poäng.

Kunskap och handling

Recommended for: **Lectures:** ca 30 h
Kursen vänder sig till informatik-doktorander vid IHH och informationssystemutvecklings-doktorander från IDA. Kursen är också öppen för andra doktorander vid IHH, IDA och CMTO, Linköpings universitet.

The course last ran:
New course.

Goals:
Kursen syftar till att öka förståelsen av samspelet mellan kunskap och handling, d v s olika kunskaps- och handlingsteorier.

Prerequisites:
Inga särskilda krav.

Organization:
Föreläsningar, seminarier.

Contents:
Pragmatisk kunskapsteori

Tyst kunskap, rekonstruktion av tyst kunskap

Praktisk kunskap

Samspel kunskap - handling

Handlingsteorier

Handlingar intentionalitet och flerfunktionalitet

Värde och handling

Reflektion och handling

Praktikteori, praktikbegreppet

Professioner, tysta kunskaper och kompetent handlande - yrkeskunnande

Literature:
B Rolf: Profession, tradition och tyst kunskap, Nya Doxa

B Molander: Kunskap i handling, Daidalos

G Goldkuhl: Praktikteori som forskningsgrund, Rapport IHH/CMTO

Ytterligare rapporter

Examiner:
Göran Goldkuhl.

Schedule:
Kursen genomförs under våren 1998, eventuellt sker kursstart i dec 1997.

Examination:
Skrivande av rapport som seminariebehandlas.

General Graduate Courses Spring 1998

Arrangörer:
Kursen samarrangeras av Internationella Handelshögskolan (IHH) i Jönköping och Institutionen för datavetenskap (IDA) och Centrum för studier av Människa, Tenik och Organisation (CMTO), Linköpings universitet.

Principles of Modern Database Systems

Recommended for:
Graduate students.

Lectures: 30 h

Appropriate for ECSEL as Comprehensive Course.

The course last ran:
Spring 1997.

Goals:
To present the fundamental concepts, theories and realizations of modern database technology and systems, with a concentration on modern Object-Oriented (OO) database systems, Query Processing, and Active Databases.

Prerequisites:
Undergraduate courses in computer science.

Organization:
Lectures and practical exercises.

Contents:
- Introduction to DBMS concepts
- History DBMSs, relational, OO, network, functional
- Introduction to Object-Oriented and Object-Relational DBMSs
- OO Query Languages, OSQL
- OO architetures and implementation
- Principles of Query Processing
- Object-relational architectures and implementation
- Active Databases
- Heterogeneous Databases
- Temporal databases
- Main-memory Databases
- Real-Time Databases
- Spatial Databases

Literature:
R. G. G. Catell: Object Data Management + handouts

M. Stonebraker: Object-Relational Databases - the next great wave

Teacher:
Tore Risch

Examiner:
Professor, Tore Risch.

Schedule:
Starts February 25th, ends April 10th, 1998

Examination:
Written exam, Seminar Papers, and Small Programming Project

Credit:
4 credits.

Principles of Programming Languages and Environments

Recommended for: **Lectures:** 20 h
Foundational course for ECSEL.

Recommended for PhD students with a non-computer-science background, e.g. students from IMIE.

The course last ran:
New course.

Goals:
The course presents basic principles behind the design of programming languages.

Basic notions are language, abstraction, typing, and semantics. Basic principles of integrated and interactive programming systems are also presented.

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 systems and mechanisms.

Literature:
(Tentative)

- Kenneth C. Louden: Programming Languages, Principles and Practice. PWS Publishing

Company, Boston, 1993. (ISBN 0-534-93277-0)

- Peter Fritzson: Specifying Practical Translators using Natural Semantics and RML. (kompendium).

- Peter Fritzson: System Development Environments. (kompendium)

Examiner:

Peter Fritzson.

Schedule:

Spring 1998.

Examination:

Written examination or homework problems. A few practical programming exercises.

Active participation during lectures.

Credits:

3 p.

Strategi och ekonomisk styrning

Recommended for:
Doktorander.

Lectures: 20-25 h

The course last ran:
New course.

Goals:
Efter genomgången kurs skall doktoranden vara väl insatt i forskningsområdet strategi och ekonomisk styrning.

Prerequisites:
Grundläggande kunskaper i ekonomistyrning.

Organization:
Föreläsningar, seminarier och inlämningsuppgifter.

Contents:
Översikt av forskningsområdet strategi och ekonomisk styrning. Utgångspunkten är ett antal tidiga klassiska studier. Därefter behandlas moderna koncept som exempelvis det balanserade styrkortet.

Literature:
Nilsson, F., 1997, Strategi och ekonomisk styrning - En studie av hur ekonomiska styrsystem utformas och används efter företagsförvärv, (ak.avh.), Linköping; Linköping Studies in Science and Technology

Simons, R., 1995, Levers of Control - How Managers Use Innovative Control Systems to Drive Strategic Renewal, Boston, MA: Harvard Business School Press

Artikelkompendium

Examiner:
Fredrik Nilsson.

Schedule:
Våren 1998.

Examination:
Aktivt deltagande på seminarium och godkända inlämningsuppgifter.

Credits:
3 p.

TESTING - Problems and Techniques

Recommended for:
PhD students and industry.

Lectures: 20-30 h

Appropriate for ECSEL as Research Frontier course.

Orientation course for Master's level students (Ch, Dh).

Goals:
The objective of the course is to present a systematic and practical approach to software testing, to study the framework and structure of the testing process, and state-of-the-art testing techniques.

Prerequisites:
Foundations in computer science.

Organization:
Lectures, presentations and exercises.

Contents:
Introduction and motivation, definition of terms used in software testing, description of framework and structure of the testing process, identifying what to test, using state-of-the-art of testing techniques correctly and appropriately, knowing when to stop testing, description of the steps required to perform testing, developing testing cases that test the software adequately, quickly and efficiently.

Litterature:
To be decided.

Teachers:
Mariam Kamkar and invited speakers.

Examiner:
Mariam Kamkar.

Schedule:
April 98.

Examination:
Will be determined later.

Credits:
4 + 2 credits.

Utredningsmetodik och kvantitativa metoder

Recommended for:
Alla doktorander. (Erbjuds för IMIE)

Lectures: 50-70 h

The Course last run:
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.

Litterature:
Mårtensson, B. Nilstun, T., Praktisk vetenskapsteori, Studentlitteratur 1998

Lekvall, Wahlbin, Information för marknadsförare

Miser, H.J., Quade, E.S., Handbook of system analysis, John Wiley & Sons 1988

Ruist, E., Modellbyggnad för empirisk analys, Studentlitteratur 1990

Savén, Produktionssimulering

Sellstedt, B., Samhällsteorier, Studentlitteratur 1992

Vedung, E., Utvärdering i politik och förvaltning, Studentlitteratur 1992

Kurskompendium

Modifierat program

Examiner:
Birger Rapp.

Schedule:
Våren 1998 (april) med fortsättning under hösten 1998.

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

Credits:
10 p.

If enough participants show interest these courses will be given

Cognitive Anthropological Linguistics

Recommended for:
Graduate Students.

Lectures: 36 h

The course last ran:
New course.

Goals:
To introduce students to the field of cognitive anthropology and cognitive linguistics.

Prerequisites:
The equivalent of a linguistics course in the Computer Science program.

Organization:
The course is organized as a series of seminars.

Contents:
The main topics covered by the course are cognitive schemata, cultural models and folk theories that constitute the grounds for everyday actions (verbal and non-verbal), thoughts, and feelings.

Literature:
Wierzbicka, Anna (1996) Semantics Primes and Universals. Oxford: Oxford Univ. Press, ISBN 0-19-870003-2.

Examiner:
Richard Hirsch.

Schedule:
Fall term 1997.

Examination:
Course essay and active participation in course seminars or, oral or written exam.

Credit:
Up to 5 points.

If enough participants show interest these courses will be given

Cognitive Grammar

Recommended for:
Graduate Students.

Lectures: 36 h

The course last ran:
New course.

Goals:
To introduce students to the field of cognitive linguistics and cognitive grammar.

Prerequisites:
The equivalent of a linguistics course in the Computer Science program.

Organization:
The course is organized as a series of seminars.

Contents:

The main topics covered by the course are cognitive skills, cognitive domains, things, processes, cognitive relations, valence, symbolic units, categorization, context, composition, and mental spaces.

Literature:
Johnson, Mark (1987) The Body in the Mind. Chicago: Chicago Univ. Press, ISBN 0-226-40318-1.

Ungerer. F. & Schmid, H.-J. (1996) An Introduction to Cognitive Linguistics. London: Longman, ISBN 0-582-23966-4.

Langacker, R. (1987) Foundations of Cognitive Grammar vol 1. Stanford: Stanford Univ. Press, ISBN 0-8047-1261-1.

Examiner:
Richard Hirsch.

Schedule:
Fall term 1997.

Examination:
Course essay and active participation in course seminars or, oral or written exam.

Credit:
Up to 5 points.

If enough participants show interest these courses will be given

Distributed Real-Time Systems

Recommended for: **Lectures:** 48 h

Graduate Students in Computer Science.

Graduate Schools ECSEL, ARTES.

The course last ran:
Fall 1996.

Goals:
To provide participants with a thorough knowledge and understanding of the engineering and design principles required in order to design and implement large, complex, reliable distributed real-time systems. To practice some of these principles in an actual real-time environment.

Prerequisites:
Programming Methodology, Software Engineering, Distributed Systems, Real-Time Systems.

Organization:
Lectures and seminars. (Lectures will not be in Linköping). Optional programming assignment. (Lectures and seminars available via ISBN-based video conference.)

Contents:
Software Quality Attributes relevant to Distributed Real-Time Systems

Distributed Real-Time and Dependability Concepts & Models

Real-Time Communication Networks and Protocols, Abstract R-T LAN

Scheduling in Distributed Real-Time Systems, Dynamic vs. Static

Design of Distributed Real Time Systems, Time vs. Event Triggered

Programming Assignments on Reliable Distributed Real-Time Programming

Literature:

Barbacci, M.R. Lecture Notes on Software Quality Attributes. 1995. (Available in PostScript to students via WWW).
I Mullender, S. (ed.) Distributed Systems (second edition), Addison-Wesley 1993. ISBN 0-201-62427-3. Chapters 16-19.

A number of research articles (availabel from library or collection).

Examiner:
Sten F. Andler.

Schedule:
Fall 1997, (Sep 2 - Dec 16, Tuesdays 09.00-12.00).

Examination:
Seminar presentations, examination paper.

Two optional programming assignments.

If enough participants show interest these courses will be given

Credit:
4 + 2 points.

Note:
Additional information about this course can be found on WWW. Follow the pointer to the Distributed Real-Time Systems course from my "official" hom page:
http://www.his.se/ida/staff/sten.chtml
When you arrive at the home page of the course, please follow the link to "Course Description". This will contain detailed information about the contents of the course and suggestions of articles to be studied (subject to change).

If enough participants show interest these courses will be given

Geographical Information Systems - GIS

Recommended for:
All.

Lectures: ca 35 h

The course last ran:
New course.

Goals:
Give insight in Geotemporal Informatics and Geographical Information Systems - GIS. GIS is the common name for many types of information systems that combine data in time and space. Application areas within the environmental informatics, medical informatics, transport planning, "business GIS", etc. The goal of the course is to give an understanding for the principles in GIS and its components. The course will give both informatic and computer perspectives to the subject.

Prerequisites:
No special prerequisites.

Organization:
The course is held during the spring 1998 with lectures, seminars and litterature studies. Every PhD student is expected to write a scientific paper (about 15 pages) based on the application of GIS in his/her own project or after consultance with the examiner.

Contents:
Space and time are two components that in greater extent are included in models to describe natural as well as technical and social systems. In this course the specific conditions are stressed that are rised when handling geospatial/temporal information - as in GIS. Different forms of data capture conceptual models, representation, data/data base methods and analytical tools for geospatial information will be discussed. By increasing the number of dimentions, that is handled in the information system, it is possible to study dynamic processes and complex phenomena - by using data with a locality or geographic distribution korological studies can be made between different levels of functional systems.Literature:

Litterature:
Fundamentals of Spatial Information Systems, Robert Laurini & Derek Tompsson, APIC Series Nr 37, Academic Press, 1995.

Symbolic Projection for image information retrieval and spatial reasoning, S. -K. Chang & Erland Jungert, Academic Press 1996.

Teachers:
Erand Jungert, Åke Sivertun, and guest professors.

Examiner:
Åke Sivertun.

Schedule:
April - May 1998.

Examination:
A written paper on related topic, participation in lectures and participation in presentations.

If enough participants show interest these courses will be given

Credit:
5 credits.

If enough participants show interest these courses will be given

Principles of Knowledge Representation

Recommended for: **Lectures:** 12 h
Foundational course for ECSEL.

Recommended for PhD students with a non-computer-science background, e.g. students from IMIE.

The course last ran:
New course.

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 caracterize 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
Collection of articles.

Examiner:
Erik Sandewall.

Schedule:
To be determined.

If enough participants show interest these courses will be given

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

If enough participants show interest these courses will be given

Systems Modeling

Recommended for: **Lectures:** 24 h
Graduate Students.

Probably appropriate for other graduate schools (ECSEL, IMIE etc).

The Course last run:
Previously given in 1994/95.

Goals:
To introduce the formal concept of model. To introduce basic modelling techniques used in engineering and to present some details of discrete-event modelling and hybrid systems modelling.

Prerequisites:
Basic knowledge of automata theory, basic knowledge of differential equations.

Organization:
Lectures given by Jacek Malec and seminars prepared by the course participants.

Contents:
- what is a model?
- continuous vs. discrete models;
- some mathematics;
- hybrid systems;
- discrete-event systems.
- some fancy modelling techniques (cellular automata, chaos, ...)

Litterature:
Articles and lecture notes distributed during the course; alternatively a textbook.

Examiner:
Jacek Malec.

Schedule:
Fall 1997 (Oct-Dec).

Examination:
Term paper.

Credits:
3 p.

Recommended Master Courses

C3-courses

TDDA14	**AI Programming**
TDDB13	**Concurrent Programming**
TDDA37	**Compiler Construction**
TDDB38	**Database Technology**
TDDA41	**Logic Programming**
TDDA43	**Programming Theory**
TDDB42	**Semantics of Programming Languages**
TDDB61	**Methodology of Program Development and Programming Development Project**
TGTU04	**Leadership**

C4-courses

TDDA12	**System Development**
TDDA16	**Representation of Knowledge in AI**
TDDA18	**Natural-Language Processing**
TDDA32	**Design and Analysis of Algorithms**
TDDB34	**Object-Oriented System Development**
TDDB66	**Expert Systems-Methods and Tools**
TDDA67	**Distributed Systems**
TDDB02	**Software Quality**
TDDB13	**Human-Computer Interaction**
TDDB15	**Computer Aided Software Engineering for Development and Maintenance**
TDTS41	**Computer Networks**
TDTS51	**Advanced Computer Architecture**

SVP-courses

HIIC66	**Objektorienterad systemutveckling**
HIIC63	**CASE**
HIID68	**Dokumenthantering**
HIIC62	**Datorteknik och datornät**
HIIC65	**Samhällsvetenskaplig kunskapsbildning**
HIIC69	**Människa-Datorinteraktion**
HIID61	**Objektorienterad verksamhetsutveckling**
HIID62	**Programvarukvalitet**
HIIC73	**Expertsystem**
HIID64	**Design**
HIIC72	**Ledarskap**
HIID63	**Forskningsstrategi inom informationssystem**

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:

The Laboratory for Complexity of Algorithms (ACTLAB)
Fixed time for lab meetings: **Wednesdays 13-15.**

The Laboratory for Application Systems (ASLAB)
Fixed time for lab meetings: **Thursdays 13-15.**

The Laboratory for Computer Aided-Design of Digital Systems (CADLAB)
Fixed time for lab meetings: **Thursdays 13-15.**

The Laboratory for Engineering Databases and Systems (EDSLAB)
Fixed time for lab meetings: **Thursdays 13-15.**

Economic Information Systems (EIS)
No fixed time for lab meetings.

Embedded Systems Laboratory - ESLAB
Fixed time for lab meetings: **Wednesdays kl 10-12**

The Laboratory for Library and Information Science (LIBLAB)
Fixed time for lab meetings: **Wednesdays 13-15.**

The Laboratory for Intelligent Information Systems (IISLAB)
Fixed time for lab meetings: **Wednesdays 13-15.**

The Knowledge Processing Laboratory (KPLAB)
No fixed time for lab meetings.

The Laboratory for Logic Programming (LOGPRO)
Fixed time for lab meetings: **Wednesdays 15-17.**

People, Computers and Work (MDA)
Fixed time for lab meetings: **Wednesdays 13 - 15.**

Laboratory-oriented Courses and Activities

The Laboratory for Natural Language Processing (NLPLAB)
Fixed time for lab meetings: **Wednesdays 15-17.**

The Laboratory for Programming Environments (PELAB)
Fixed time for lab meetings: **Thursdays 15-17.**

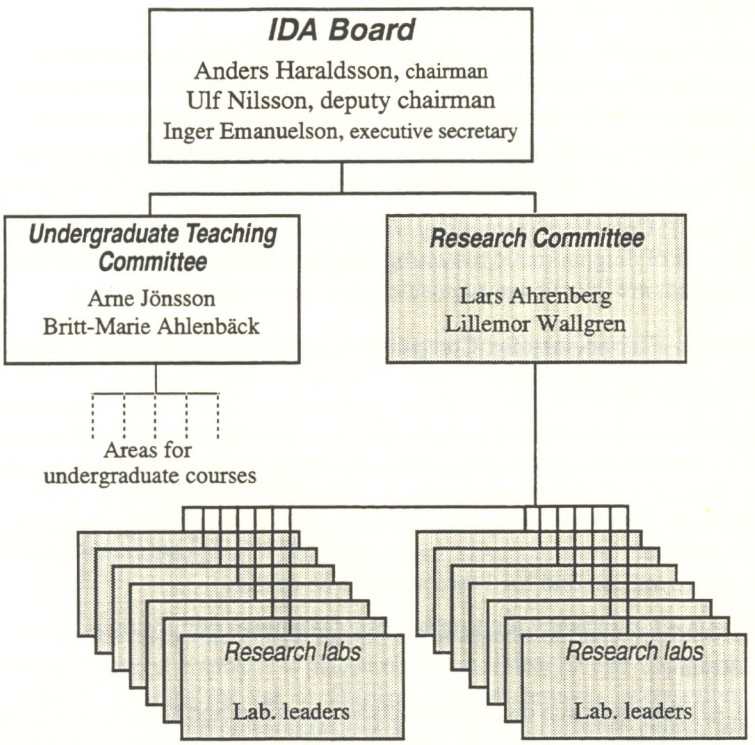
The Real-Time Systems Laboratory (RTSLAB)
Fixed time for lab meetings: **Wednesdays 13-15.**

The Autonomous Systems Laboratory (TASLAB)
No fixed time for lab meetings.

The Laboratory for Temporal-System Correctness and Algorithms (TOSCA)
No fixed time for lab meetings.

Information Systems and Work Contexts (VITS)
Fixed time for lab meetings: **Some Mondays 10-12.**

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 17 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 below.

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.

ASLAB – Application Systems Laboratory

Sture Hägglund and Claes Wohlin

ASLAB/Promise: Industrial Software Engineering, in particular software process improvement, software reliability, requirements engineering, object-orientation and empirical studies of development and maintenance of large-scale industrial software systems.

ASLAB/UM: Usability Matters, with a focus on usability engineering of interactive systems. Research areas include methods, tools and innovative concepts for interaction design.

ASLAB/CT: Cognition Technology, with studies of knowledge-based decision support, cognitive models of diagrammatic reasoning, intelligent tutoring and simulation for training. Special interest areas are knowledge acquisition and web-based architectures.

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 database 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.

EIS – Economic Information Systems

Birger Rapp

EIS has six professors (one full time and five part time). 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 devlopment of structures within organisations, together with the effects of information technology on people and organisations. 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 knowledge representation/organization and processing 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 of the representation and processing of knowledge. Special emphasis is placed on the specification and implementation of agent-oriented software systems for use in distributed, heterogeneous information landscapes. Current activities include the development of interface and software agents used for gathering, processing, and filtering hypertext based information on the Internet and WWW. Other activities of interest are visual and temporal reasoning, planning, hypermedia, and computer-supported education.

LIBLAB – Laboratory for Library and Information Science

Roland Hjerppe

LIBLAB studies methods for access to documents and the information contained in the documents, concentrating on catalogues and bibliographic representations, and on the human factors of library use. Current interests are focused on i.a. document architecture issues, the merging of information from libraries, archives and museums, spatio-temporal information and geographic information systems, and formal approaches to the analysis of qualitative data.

LOGPRO –Logic Programming Laboratory

Jan Maluszynski and Ulf Nilsson

Research in LOGPRO focuses on the relationship between logic programming and other programming paradigms and specification languages. The current research concentrates on (1) static analyses and query optimization techniques in logic programs and deductive databases (2) verification of program properties and (3) declarative dubugging of constraint programs. The long-term goal of the research is to improve the efficiency and ease of use of declarative languages.

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 inter-disciplinary 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 language engineering systems from linguistic, computational and behavioral perspectives. Current projects concern natural-language dialog systems, computer-aided translation and language engineering for information retrieval.

PELAB – Programming Environments Laboratory

Peter Fritzson

PELAB is concerned with research in 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 debuggning 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 scientific computing, program transformation systems, 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 of control and supervisory functions, and multi-level architectures for real-time software. 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 Algorithms

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 modelling and verification of embedded systems. We use hybrid mathematical models (mixed continuous and discrete), as well as discrete (synchronous) languages for modelling purposes.

VITS - Development of Information Systems and Work Contexts

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, inter-organizational information systems.

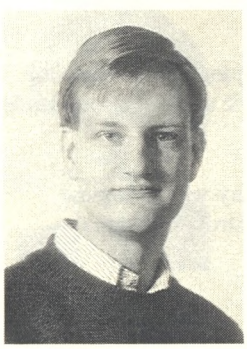
Faculty engaged in the graduate study programme

Leif Appelgren, Lic. Consulting professor (*adjungerad professor*), economic information systems. Previous academic appointments with Linköping and Handelshöyskolen Bl, Norway.

Several previous affiliations.

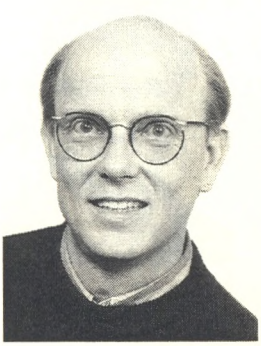
Lars Ahrenberg, Ph. D., Uppsala 1987. Professor of computational linguistics. Group leader, NLPLAB. Research Committee chairman.

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.



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.

Theoretical and practical aspects related to the representation and processing of knowledge. Specific areas of interest are agent-oriented software systems, visual reasoning, and innovative learning technologies.

Wlodzimierz Drabent, Ph. D., Warsaw 1985. Part time associate professor (*docent, universitetslektor*), computer science. Position 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 (*universitetslektor*), logic and AI. Group leader, TASLAB.

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

Henrik Eriksson, Ph. D., Linköping 1991. Associate professor (*docent, universitetslektor*), computer science. Previous affiliation: Stanford University, Stanford, CA, 1991-94. Part time leave to SICS 1997.

Knowledge-based systems, knowledge acquisition, medical informatics, software development environments, software reuse.

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

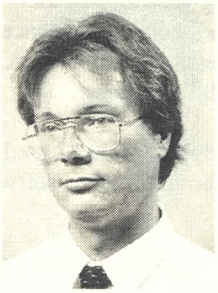
Distributed systems, object-oriented programming, object-oriented analysis and design, operating systems.

Thomas Falk, Ph. D., Stockholm School of Economics 1976. Consulting Professor (*adjungerad professor*), economics of information technology. Several previous affiliations. Current affiliation: Industrial policy 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, analysis of algorithms, data structures.

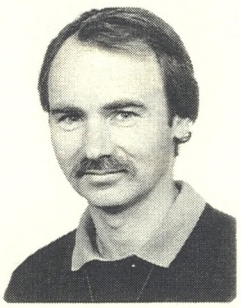


Dan Fritzson, Ph.D., Göteborg 1988. Consulting professor (*adjungerad professor*), engineering information systems especially scientific computing. Previous affiliation Chalmers, Göteborg. Current affiliation SKF.

Bearing simulation, parallel simulation, modeling visualization.

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, scientific computing, debugging tools, incremental compilation technology, compiler generation, compilers and development tools for parallel hardware.



Göran Goldkuhl, Ph. D., Stockholm 1980. Acting professor of information systems development. Professor of informatics, Jönköping International Business School. Group leader, VITS. 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.

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. Assistant professor (*universitetslektor*), ASLAB. Assistant professor (forskarassistent), IKP Kvalitetsteknik. Previous affiliations: Northeastern University, Boston, IBM Corporation.

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

Roland Hjerppe, senior lecturer. Group leader, LIBLAB. Previous affiliations: KTH, DFI and expert mission Tanzania. Visiting Distinguished Scholar at Office of Research, OCLC Inc. in Columbus, Ohio, 1988-89.

Library science and systems, hypertext and -media, knowledge organization and information retrieval, citation analysis and bibliometrics, computer support for personal and cooperative activities, virtual environments.

Sture Hägglund, Ph. D., Linköping 1980. Professor of knowledge-based systems. Deputy head of the department of computer science. Research Committee chairman 1990 - 1997. 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. Assistant professor (*universitetslektor*), computer science.

Construction and analysis of algorithms. Complexity theory.

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

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

Mariam Kamkar, Ph. D., Linköping 1993. Assistant professor (*universitetslektor*), computer science. 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 (*vik. 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 coumputer science 1983-90. Previous affiliation: Carlstedt Elektronik AB, Göteborg 1992-94.

Software engineering, real-time systems, industrial evaluation of elements of new software technology.

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

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

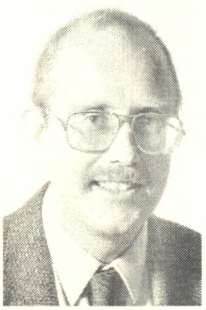
Reactive systems, autonomous systems, system theory, knowledge representation, artificial intelligence.

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

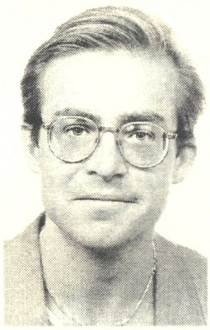
Logic programming, formal language theory, integration of programming paradigms.

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.



Anders G. Nilsson, Econ Dr., Stockholm 1991. Consulting professor (*adjungerad professor, docent*), information systems development. Acting professor of economic information systems in 1992. Position at the Institute for Business Process Development (Institute V) and Stockholm School of Economics.
Application packages, business modelling, business process reengineering (BPR), information management, ISD methods, IS/IT strategies, maintenance management.



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; Evaluation strategies for query processing; program transformation and abstract interpretation.

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

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

Kjell Orsborn, Ph. D., Linköping 1996. Assistant professor (*forskarassistent*), engineering information systems.

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.

Mikael Pettersson, Ph. D., Linköping 1995. Assistant professor (*forskarassistent*), computer science especially programming language implementation.

Programming language semantics and implementation, computing systems infrastructures and engineering.

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.

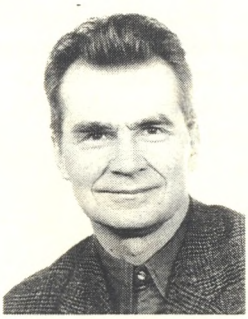
Birger Rapp, Econ. Dr., Stockholm 1974, Professor of economic information systems. Group leader, EIS. Vice president at large for IFORS. Editorial (advisory) boards to EJOR, IJMSD, JORBEL and Omega. President of the Pronova Research and Development Board in Sweden.

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.

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.

Nahid Shahmehri, Ph. D., Linköping 1991. Associate professor (*docent*, *universitetslektor*), computer science. Group leader, IISLAB. Director of graduate study programme.

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.

Lena Strömbäck, Ph.D., Linköping 1997. Assistant professor (*vik. universitetslektor*), computational linguistics.

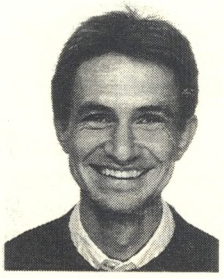
Natural language understanding, tools for grammar development, unification-based formalisms, models for natural language.

Toomas Timpka, MD., Stockholm 1983, Ph. D., Linköping 1989. Associate professor (*docent, universitetslektor*), computer and information science. Group leader, MDA. Acting Professor of Social Medicine and Associate Professor of Medicial 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.



Claes Wohlin, Ph. D., Lund 1991. Professor of software engineering. Part time at Lund university. Previous affiliations: Lund University and 5 years of industrial experience in software engineering in 1988-93.

Experimental software engineering, software quality, system validation, software reliability, process improvement and requirements engineering.

Guest researchers and affiliated faculty engaged in the graduate study programme

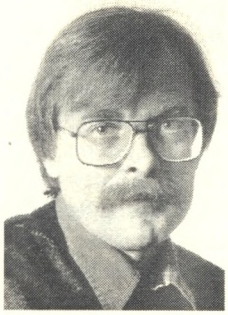
Sten F. Andler, Ph. D., Carnegie-Mellon University 1979. Professor of computer science, Skövde university College, (*docent* LiU), distributed systems, real-time systems, operating systems. Previous affiliations: IBM Software Solutions, San Jose, CA, 1992-93, IBM Almaden Research Center, San Jose, CA, 1979-92. Group leader for Distributed Real-time Systems (DRTS) research group. Distributed real-time systems, real-time databases, active real-time databases, distributed databases, real-time operating systems.

Staffan Bonnier, Ph. D., Linköping 1992. Assistant professor (*universitetslektor)*, computer sience especially real-time systems. Previous affiliation for Carlstedt Elektronik AB, Göteborg 1993-1994. On leave up to fall 1998.

Formal methods in real-time system development, theory and practice of declarative programming.

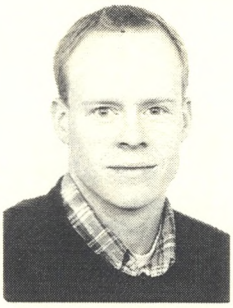
Witold Litwin, Professor University Paris 9. Guest professor Linköping, Stanford, Berkeley, Santa Clara univ.

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



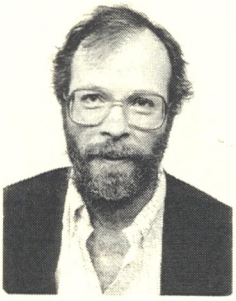
Witold Lukaszewicz, Ph.D.,Warsaw University 1979. Guest professor. On leave from the Institute of Informatics, Warsaw University, Poland.

Knowledge representation, non-monotonic reasoning, programming methodology.



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

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

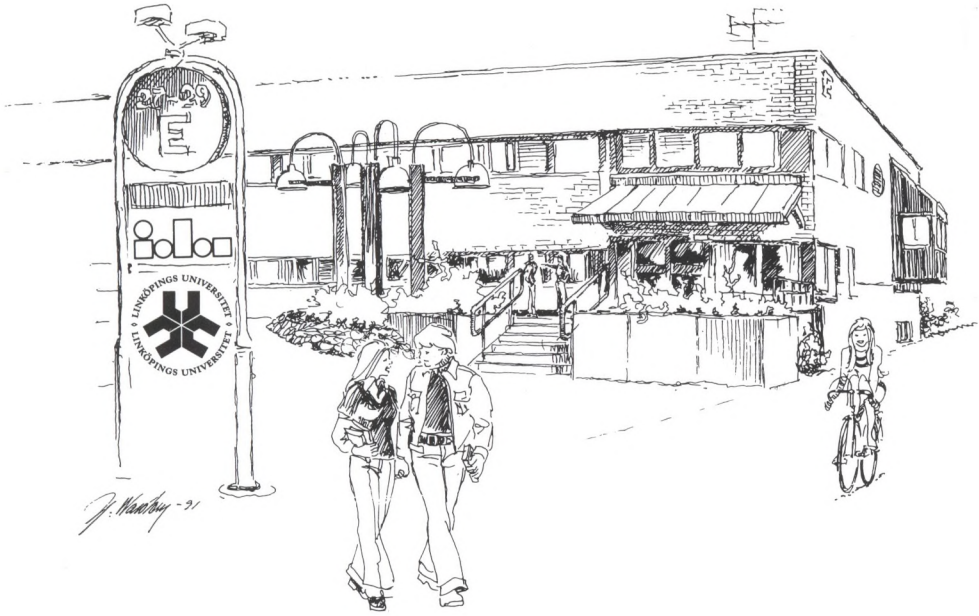


James M. Nyce, Ph.D., Brown 1987. Guest professor, computer and information science. Docent, information science. Previous affiliation Brown.

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

Kristian Sandahl, Ph. D., Linköping 1992. Associate professor *(docent, universitetslektor)*, computer science. On leave from 1996 to Ericsson Radio Systems, Linköping.

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


LINKÖPINGS UNIVERSITET
LINKÖPINGS UNIVERSITET

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