DEPARTMENT OF COMPUTER AND INFORMATION SCIENCE (IDA)

INTERNATIONAL GRADUATE SCHOOL IN COMPUTER SCIENCE

COURSES 96/97

Spring

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**
- **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 16 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

LINKÖPINGS UNIVERSITET ◇ LINKÖPINGS UNIVERSITET ◇

Graduate Courses Spring 1997

Course	Course literature	Author	Jan	Febr	March	April	May	June	Day	Time	Room
Automated Reasoning Credit 5 p Andrei Voronkov	A Compendium is prepared.		Starts Jan 31		Ends March 21				Jan 31 Feb 14,21 Mar 7,21	10-17	Belöningen
Compiler Construction - Advanced Course Credit Part I 1 p Credit Part II 4 + 1-2 p Mariam Kamkar	Not yet decided		Part I: Starts Jan 28	Part I: Ends Febr 18		Part II: Starts Apr 22	Part II: Ends May29		Part I: Tue Part II: Tue Thur	10-12 9-12 9-12	Belöningen
Compiling Functional Languages Credit 3 p (+ 2 p) Mikael Pettersson	The Implementation of Functional Programming Languages	Simon L. Peyton Jones				Starts April 25		Ends June 17	Fri Tues	13-16 15-17	Belöningen Estraden
Constraint Logic Programming Credit 4 + 2 p Wlodek Drabent, Jan Maluszynski	Selected survey and research papers				Starts March 14		Ends May 23		Fri (May 16, 23) Tue	13-15, (13-17) 15-17	Elogen
Decision Support for Management Credit 3 p Sven Carlsson				Starts Febr 5							Eliten
IT-Ekonomi och Informationsekonomi Credit 3 p Nils-Göran Olve	1. IT som strategisk resurs 2. Investing in Information Technology	1. Falk & Olve 2. Hogbin & Thomas	Separate Schedule								
The Personal Software Process Credit 5 p Claes Wohlin	A Discipline for Software Engineering	Watts S. Humprey	Starts Jan 8				Ends May 28		Wed	10-12	Estraden
Principles of Modern Database Systems Credit 4 + 2 p Tore Risch	1. Object Data Management 2. Object-Relational Databases	1. R.G.G. Catell 2. M. Stonebraker:		Starts Febr 25		Ends Apr 17			Tues Thur	10-12 10-12	Belöningen (except v 12 - Estraden)
Programming 3D Graphics and Virtual Reality Credit 3 p Peter Fritzson	Articles and reports. Parts of Open GL graphics standard						Starts May 12	Ends June 16	Mon	13-16	Elogen
Speech Technology Credit 5 p Lars Ahrenberg	Voice comunication with computers	Chris Schmandt	This is an undergraduate course - Separate Schedule								Belöningen
Synthesis Methodologies for Digital System Credit 3 p Krzysztof Kuchcinski	Kris is writing a book				Starts Mar 17		Ends May 9		w. 12-17 Mon w. 12-19 Fri	13-15 10-12	Eliten
UNIX Internals - Advanced Course in Operating Systems 3 + 1 p Lars Viklund	Unix Internals, the New Frontiers	Uresh Vahalia			Mar 3, 10	Apr 28 Apr 7,14	May 5		Mon Mon	9-12 13-16 9-12	Estraden
Utredningsmetodik och Kvantitativa Metoder Credit 10 p Birger Rapp			Separate Schedule Starts in April								

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, Library and Information Science, Economic Information Systems, Computational Linguistics, Engineering Information Systems and Information Systems Development.

The department also participates in two new graduate schools aiming for interdisciplinary studies preparting also for a career outside the university. 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 Indusrtrial 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 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). About 10 of the students and about 25% of the teachers in the graduate programme have foreign citizenship or origin, which makes the programme activities very international and English the language of the programme.

The programme leads to one of the following degrees:

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

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

The Research Committee, headed by Sture Hägglund, 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 for the graduate students, and for the technical and administrative staff. 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 graduate students are employed by the department, full time. Their responsibilities comprise of, for example, assisting 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 1996/1997

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 above mentioned courses the following activities take place in the International Graduate School of Computer Science and are strongly recommended for all graduate students:

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, Per-Olof Fjällström, tel. 013-28 24 12, Britt-Inger Karlsson, tel. 013-28 17 06 or for a particular course from the person responsible for the course.

Linköping, June 20, 1996

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

Aspects of Scientific Writing

Recommended for:

Everyone

Lectures: 10 h

The course last ran:

1995/96

Goals:

To provide bases for writing scientific papers and theses. To increase the awareness of problems (and possibilities) in the writing process. To provide hints on how to improve the style of writing, how to organize a paper and how to take advantage of typography in order to communicate the content.

Prerequisites:

None

Organization:

Five seminars and two group sessions.

Contents:

- Scientific writing: what is it?
- The writing process
- Style and content
- Typography
- Writing a thesis (case studies)
- Tools for writing

Literature:

Jarrick and Josephson, *Från Tanke till Text,* Studentlitteratur

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

Teachers:

Ulf Nilsson plus invited lecturers

Examiner:

Ulf Nilsson

Schedule:

Sept-Oct 96

Examination:

Exercises and active participation

Credit:

Up to 3 credits

Note:

The seminars vill be given in Swedish

Distributed Real-Time Systems

Recommended for:

Lectures: 45 h

Graduate Students in Computer Science

The course last ran:

New course in Linköping. Has run as an M.Sc. module in Skövde 1994-95.

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 practise some of these principles in an actual real-time environment.

Prerequisites:

Programming, software engineering, distributed systems, real-time systems.

Organization:

Lectures + presentations + programming assignments + self study
(Lectures and presentations available via ISDN-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 Real-Time LAN
Scheduling in Distributed Real-Time Systems, Dynamic vs. Static
Design of Distributed Real Time Systems, Time/Event-Triggered Paradigms
Programming Assignments on Reliable Distributed Real-Time Programming

Literature:

Barbacci, M.R. Lecture Notes on Software Quality Attributes. 1994-1995.
Mullender, S. (ed.) Distributed Systems (2nd edition), Addison-Wesley 1993.
Collection of articles

Teachers:

Sten Andler, Mario Barbacci

Examiner:

Sten Andler

Schedule:

Sept-Dec 1996

Examination:

Two examination papers, seminar presentations, and two programming assignments.

Credit: 4 + 2 poäng

Företagsvärdering

Recommended for:

Ekonomiska informationssystem.

Lectures: 20-30 h

The course last ran:

Hösten 1994.

Goals:

Efter genomgången kurs skall deltagarna självständigt kunna genomföra en företagsvärdering. Det innebär kunskaper om olika värderingsmetoder, substansvärde, avkastningsvärde samt kassaflödesmodeller, vanliga analysproblem samt kapitalkostnadsberäkning.

Prerequisites:

Ekonomisk grundexamen.

Organization:

En betydande del av kursen kommer att bestå av tillämpningsövningar där deltagarna får arbeta med exempel på värderingar. Därtill kommer föreläsningar kring metodproblem och genomgångar.

Literature:

Huvudbok: Copelane et al. Valuatio, Wiley (p 427)
Olve, N-G, Företag köper företag, Mekanförbundet, p 187
Artikelsamling
Ev ytterligare litteratur

Teachers:

Prof. Rolf Rundfeldt.

Examiner:

Prof. Rolf Rundfeldt.

Schedule:

Start slutet av augusti 1996, slut december 1996.

Examination:

För godkänt på kursen krävs ett aktivt deltagande i övningar samt genomförda övningsuppgifter.

Credit:

5 poäng.

Introduction to Research Methodology in Computer Science

Recommended for:

New graduate students.

Lectures: 16 h

The course last ran:

Fall 1995.

Goals:

To give an introduction to the philosophy of science, the special characteristics of computer science research and to discuss practical 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. Project management.
Research paradigms in computer science.

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:

September - November 1996

Examination:

Written examination and seminar activity.

Credit:

2 points

Internet methodology

Recommended for:

Everyone.

Lectures: 24 h

The course last ran:.

New course.

Goals:

To make the students semi-0professionals in using, contributing to, and understanding all the resources of Internet, and issues surrounding it, especially as they relate to their studies.

Prerequisites:

Easy access to Internet.

Organization:

The course is made up of seminars and exercises to be carried out between the seminars.

Contents:

The course has the following major foci:
- protocols etc
- resources and services, in general and in C&IS
- finding,
- contributing,
- organizing,
- issues.

Literature:

All the material used will be available on Internet, lists of URLs will be provided as the course proceeds.

Teachers:

Roland Hjerppe, guests.

Examiner:

Roland Hjerppe

Schedule:

Sept-Nov 1996, once a week.

Examination:

A WWW-document about Computer and Information Science and Internet.

Credit:

3 p.

Informationsteknik och Management

Recommended for:

Forskarstuderande inom IDA, IMIE och EKI

Lectures: 36 h

The course last ran:

1995

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 1996 . Den består av föreläsningar, litteraturstudier, seminarier, skriftlig bokrecension samt en av varje doktorand författad litterturbaserad 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 allt större 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 arbetssätt och affärsprocesser.

Literature:

Artiklar och bokavsnitt.

Teachers:

Thomas Falk

Examiner:

Thomas Falk

Schedule:

Oktober och december 1996.

Examination:

Skriftlig tentamen, skriftlig bokrecension, godkänd uppsats samt opposition.

Credit:

5 poäng

Parallel Programming in Languages and Techniques

Recommended for:

Lectures: 36 h

The course last ran:

1993/94

Goals:

To give an understanding of different parallel programming models and compilation techniques for several kinds of programming languages.

Prerequisites:

he undergraduate compiler courses Compiler Construction (TDDA37) or Compilers and Interpreters (TDDA28), or equivalent.The process programming course.

Organization:

Contents:

Parallel execution models, languages, etc. For example: definition of parallel computing, measures of performance, parallel processors, shared-memory parallel programming, distributed-memory parallel programming, data parallel programming, functional dataflow programming, scheduling parallel programs, parallel programming support environments. Message Passing Interface - MPI, Parallel C++, High Performance Fortran. Practical programming exercices on a parallel machine.

Literature:

Ian Foster: Design and Building Parallel Programs. 1995, Addison-Wesley.
+ Articles

Teachers:

Peter Fritzson and invited lecturers

Examiner:

Peter Fritzson

Schedule:

Sept-Dec 1996

Examination:

Written examination.
Implementation project gives extra 1-2 points.

Credit:

5p (+1 or 2p)

Perspectives in Cognitive Science

Recommended for:

Graduate students.

Lectures: 20 h

The course last ran:

Occasional course.

Goals:

Presentation and discussion of the current front-line research topics in cognitive science.

Prerequisites:

Basic course in cognitive psychology.

Organization:

Seminars and lectures. Term paper. Possibility for individual assignments.

Contents:

Present and discuss the nature of Expertise, both technical and lay expertise. Comparison and evaluation of theories and methods from Behavioral science to study complex Decision making. "Hands-on" details for study design.

Literature:

To be distributed at the course start.

Teachers:

Vimla Patel PhD, McGill University

Examiner:

Vimla Patel PhD (/Toomas Timpka MD PhD)

Schedule:

September-November 1996.

Examination:

Attendance at seminars. Term paper.

Credit:

3 p.

Semiotics: History, Basic concepts and Computer Applications

Recommended for:

Graduate students.

Lectures: 36 h

The course last ran:

Fall 1991.

Goals:

The purpose of the course is to present the concepts of the general theory of signs and their historical development within different traditions of investigation. With a firm grasp of the theoretical concepts in hand, various concrete computer applications will be presented and discussed.

Prerequisites:

None.

Organization:

12 three hour seminars about once every two week covering the history and basic concepts of semiotics followed by supervised individual study towards the end of the course in preparation for the presentation of term papers.

Contents:

The course consists of two parts. The first part consists of a historical survey of the development of the key concepts of semiotic theory and a comparison of different traditions of research within semiotics. The concepts of sign, icon, index and symbol constitute the focus for the historical, theoretical, and comparative part of the course. the syntactic, semantic, and pragmatic dimensions of semiotic systems are also defined, explained, and exemplified. The second part of the course consists of independent supervised investigations by the participants into different aspects of the theoretical concepts and their relevance for applications in computer science.

Literature:

Andersen, P.Bl (1991) A Theory of Computer Semiotics: Semiotic approaches to construction and assessment of computer systems. Cambride: Cambridge University Press

Eco, Umberto (1984) Semiotics and the Philosophy of Language. Bloomington: Indiana University Press.

Robert E. Innis (ed.)(1985) Semiotics: An Introductory Anthology. Bloomington: Indiana University Press.

Thomas A. Seboek (1994) An Introduction to Semiotics. London: Pinter Publ.

Teachers:

Richard Hirsch

Examiner:

Richard Hirsch

Schedule:

Sept-Dec 1996

Examination:

A written paper on a chosen theoretical topic or a report on an investigation of computer applications of semiotic theory.

Credit:

Up to 5 points

Automated reasoning

Recommended for:

Lectures: 30 h

Anyone.

Goals:

To introduce the area of automated deduction. To give the basic knowledge which can help in research in many areas of commputer science.

Prerequisites:

Little knowledge of mathematical logic is an advantage, but not necessary.

Organization:

12 intensive lectures + 3 exercise sessions in 5 days. The first days, there will be 3 lectures each day. The last three days, there will be 2 lectures and one exercise session every day. There will also be home exercises.

Contents:

- Introduction and history;
- Resolution-based theorem proving;
- Sequent-based methods (semantic tableaux, connection method, the inverse method);
- Reasoning with equality and other built-in relations;
- Applied reasoning (logic programming, deductive databases, model checking).

Literature:

The field suffers from the lack of good monographs. I will prepare a compendium before the course and make its corrected version along with the lectures

Teachers:

Andrei Voronkov, Uppsala University

Examiner:

Andrei Voronkov

Schedule:

Spring 1997

Examination:

Written exam or an implementation project, at the student's choice.

Credit:

5 p

Compiler Construction - Advanced Course

Recommended for:

For graduate students

Lectures:

Part 1, Lectures	8 h
Part 2, Lectures	30 h
Project lab work	10-20 h

The course last run:

New course.

Goals:

The goal of this course is to give the student a thorough insight in compiler construction. The course includes topics not included in undergraduate courses or advanced research topics.

Prerequisites:

The undergraduate compiler courses Compiler Construction (TDDA37) or Compilers and Interpreters (TDDA28), or equivalent. Pteori I is also recommended but not required.

Contents:

Part 1:

- **Attribute grammars, Denotational semantics**
  (Only for students who have not taken Pteori I (TDDA43) or equivalent)

Part 2:

- **Code generation:**
  Code generator generators, register allocation, register colouring, code generation by pattern matching
- **Data flow analysis:**
  Basic blocks, flow graphs, Intraprocedural Data flow analysis, Interprocedural Data flow analysis, reducible flow graphs, iterative data flow algorithms, monotone data flow analysis frameworks
- **Optimization:**
  Code-improving transformations, Alias analysis, Peephole optimization
- **Compiler Generation Techniques:**
- **Project: Practical programming exercises on solving data flow analysis equations**

Literature:

(not yet finally decided)
(Book by Ken Zadeck, et. al.) Articles and reports.
(M.S. Hecht: Flow Analysis of Computer Programs, North-Holland)

Teachers:

Course leader, lectures: Mariam Kamkar

Examiner:

Mariam Kamkar

Schedule:

Spring 1997

Examination:

Part 1: written examination.
Part 2: exercises and written examination, or alternatively: exercises and a written summary of the course material.

Credit:

Part 1 gives 1 point for students who have not taken Pteori I or equivalent.
Part 2 gives 4 points for the theoretical part, and 1-2 points for the project part.

Compiling functional languages

Recommended for:

For computer science and systems students

Lectures: 27 h (9 x 3)

The course last ran:

1991/92

Goals:

The course gives a basic understanding of methods for implementing strict and lazy functional languages.

Prerequisites:

Some knowledge of ML and denotational semantics, e.g. from Pteori I or the Compiler Construction - advanced course.

Organization:

Mainly lectures. Possibly a few seminars for student presentations.

Contents:

Part I: The lambda calculus, translation of a functional language into lambda calculus, types and pattern-matching, compilation of pattern-matching, transforming enriched lambda calculus, list comprehensions, polymorphic type-checking.

Part II: Graph reduction. Program representation, selecting redex, supercombinators, lambda-lifting, fully-lazy lambda-lifting, sk combinators, storage management and garbage collection.

Part III: Advanced graph reduction. The G-machine, G-code, optimizations to the G-machine, optimizing tail calls, strictness analysis, parallel graph reduction.

Literature:

Simon L. Peyton Jones: *The Implementation of Functional Programming Languages*, Prentice-Hall 1987.

Teachers:

Mikael Pettersson, + invited lecturer (Thomas Johnsson and Lennart Augustsson)

Examiner:

Mikael Pettersson

Schedule:

Spring 1997.

Examination:

Mandatory handins and a programming project.

Credit:

3 p(+2p)

Constraint Logic Programming

Recommended for:

Graduate Student

Lectures: 26 h + seminars 10 h

The course last ran:

New course

Goals:

The idea of constraint logic programming has been formulated in late eighties as an extension of logic programming. Since then the field matured a lot and resulted in a number of commercial systems which are increasingly applied in industry.

The object of the course is to give an introduction to and a survey of constraint logic programming. The students will have access to a number of CLP systems discussed during the course.

Prerequisites:

Basic familiarity with logic programming

Organization:

Lectures, invited seminars and a programming assignment

Contents:

1 Introduction to the field

2 Domain-specific aspects:
Interval CLP
Finite-domain CLP

3 A survey of CLP systems (CHIP, PROLOG IV, constraint solvers in SICSTus)

4 Case studies of selected applications, published in the literature or presented by invited lecturers

5 Introduction to concurrent constraint programming

Literature:

Selected survey and research papers

Teachers:

W Drabent, J Maluszynski, U Nilsson and invited speakers

Examiner:

W Drabent and J Maluszynski

Schedule:

Spring 1997

Examination:

Seminar, programming assignment

Credit:

4+2 points depending on the effort and quality of own contribution

IT-ekonomi och informationsekonomi

Recommended for:

Ekonomiska informationssytem.

Lectures: 34 h

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 1996 ämnen såsom: Fördelning av IT-ansvar; Överväganden inför och effekter av investering i grupprogramvara; Kostnader och nytta av IT-utbildning; IT-kompetens ur ett Balanced Scorecard-perspektiv. Arbetet kan med fördel användas för att testa tankar inför val av avhandlingsämne.

Contents:

ITs 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 gäster.

Examiner:

Nils-Göran Olve.

Schedule:

Start våren 1997.

Examination:

Godkända arbetspapper, framlagda vid seminariet.

Credit:

3 poäng.

Principles of Modern Database Systems

Recommended for: **Lectures:** 32 h

Graduate students.

The course last ran:

Fall 1995.

Goals:

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

Prerequisites:

Undergraduate courses in computer science. Basic database course preferred, but not required.

Organization:

Lectures and seminars covering the core material of the course. Extra points for programming project or seminar papers.

Contents:

- Overview Traditional DBMs
- Overview Data Models (e.g. relational, O, functional)
- Object Data Management Concepts
- Query processing (OO and relational)
- Versioning
- Database Performance and Benchmarks
- Active Databases
- Temporal Databases
- Main-memory Databases
- Real-time Databases

Literature:

R.G.G. Catell: Object Data Management + handouts
M. Stonebraker: Object-Relational Databases - the next great wave

Teachers:

Tore Risch

Examiner:

Tore Risch

Schedule:

Spring 1997.

Examination:

Written exam, Seminar Papers, and Small Programming Project

Credit: 4 + 2 p

Programming 3D graphics and virtual reality

Recommended for: Lectures: 20 h

Graduate students or master students in computer science.

The course last ran:

New course.

Goals:

To give the participants knowledge about effectives techniques for programming 3D graphics, animation and virtual reality applications.

Prerequisites:

Basic programming knowledge including some experience of C/C++ and programming window system applications.

Organization:

Lectures and seminars. Practical programming exercises.

Contents:

Models for defining graphics. Hierarchical structured graphics objects. Parametric surface techniques. Rendering and texture. Interaction with 3D objects. 3D graphics standards such as OpenGL. Animation. Virtual reality and modeling. Cooperative work within virtual reality environments. Current hardware architectures for 3D and animation.

Literature:

Articles and reports. Parts of OpenGL graphics standard, Addison-Wesley, 1995.

Teachers:

Peter Fritzson + invited speakers

Examiner:

Peter Fritzson

Schedule:

Spring 1997.

Examination:

Presentations and Programming exercises.

Credit:

3 p

Speech technology

Recommended for:

Lectures: 24-30h

Computer Science and Computational Linguistic students.

The course last ran:

New course.

Goals:

The course will provide basic knowledge of standard techniques for speech recognition and speech synthesis and of their use in applications such as user interfaces, spoken dialogue systems and speech aids.

Prerequisites:

Degree in Computer Science (C-program) of Computer Science and Engineering (D-program) or basic knowledge in either signal processing or natural-language processing.

Organization:

The course has three parts. The first part is introductory and provides basic prerequisities in signal processing or linguistics/NLP (as required). The second part is focused on speech technology and its theoretical underpinnings. Lab work with speech systems is also included.
The course will be taught inseminar form with several guest lectures.

Contents:

(Preliminary) Element of acoustics, signal processing. Elements of phonetics and phonology. Models of speech communication, speech production and perception. Speech synthesis (articulatory and LPS-techniques). Speech recognition (rule-based, stochastic, DTW-techniques). System architectures for spoken dialogue systems and speech understanding.

Literature:

A collection of articles and a basic course book to be decided on later, e.g. Keller, E. (ed.)(1994) Fundamentals of Speech Synthesis and Speech Recognition. Basic Concepts, State of the Art and Future Challenges. Chichester: Wiley.

Teachers:

Lars Ahrenberg, Arne Jönsson and invited lecturers.

Examiner:

Lars Ahrenberg.

Schedule:

January-March 1997.

Examination:

Two contributions to the course in the form of a presentation or a prepared comment on some article(s).

Credit:

5 points.

Synthesis Methodologies for Digital Systems

Recommended for:

Computer Science and computer system students.

Lectures: 24 h

The course last ran:

New course.

Goals:

The course will present the state-of the art synthesis methods applied to systemsconsisting of hardware and software parts.

Prerequisites:

Programming languages, Petri nets basics.

Organization:

The course will consist of lectures.

Contents:

Introduction
VHDL
System synthesis
Transformational design basics
Synthesis of advanced features
Modeling and synthesis of timing requirements
Hardware/Software Co-design
Test Synthesis

Literature:

A System Synthesis with VHD - A Transformational Approach, a draft of the book by course leaders.

Teachers:

Petru Eles, Krzysztof Kuchcinski and Zebo Peng

Examiner:

Krzysztof Kuchcinski

Schedule:

March-May 1997

Examination:

Obligatory homework.

Credit:

3 p

The Personal Software Process

Recommended for: Lectures: 24 h

Ph.D. students with an interest in industrial software development, who are interested in learning more about systematic process improvement through software measurement, reuse of knowledge and code in order to improve productivity and predictability of software development.

The course last ran:

New course.

Goals:

The objective is to provide knowledge in software process improvement in general, and into the personal software development process in particular.

Prerequisites:

Knowledge in at least one programming language and basic knowledge in mathematical statistics, for example, linear regression.

Organization:

Lectures and exercises to hand-in.

Contents:

The course is based on the Personal Software Process suggested by Watts D. Humphrey, SEI. The course is organized around ten programming exercises and five reports. To implement the programs, the Ph.D. students are expected to follow a predefined process which is improved continuously through seven increments. At each lecture, the material in the book is gone through, important aspect are stressed and feedback is given on the previous exercise. The feedback consists of both comments on the exercise and a summary of the measurements collected. The content is focused upon process improvement, measurement, estimation and prediction of time and size, reuse of knowledge and code, and use of your own programs within the series of exercises.

Literature:

A Discipline for Software Engineering by Watts S. Humphrey, Addison-Wesley, 1995.

Teachers:

Claes Wohlin.

Examiner:

Claes Wohlin.

Schedule:

Mid-October 1996 to Mid-March 1997.

Examination:

The ten programming exercises and the five reports should be approved.

Credit: 5 p

Utredningsmetodik och Kvantitativa metoder

Recommended for:

Alla doktorander.

Lectures: 50-70 h

The course last ran:

Spring 1996.

Goals:

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

Prerequisites:

Inga.

Organization:

Föreläsningar, seminarier, inlämningsuppgifter, tentamina.

Contents:

Vetenskapsteori, Kvalitativ undersökningsmetodik, Modelltänkande och systemanalys, Statistiska metoder, Simulering, Prognoser, Utvärdering och Presentationsteknik.

Literature:

Mårtensson, B., Nilstun, T., *Praktisk vetenskapsteori*, Studentlitteratur 1988
Lekvall, Wahlbin, *Information för marknadsförare*
Miser, H.J., Qaude, 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

Teachers:

Ett flertal

Examiner:

Birger Rapp

Schedule:

Våren och början av hösten 1997.

Examination:

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

Credit:

10 p

Advanced Course in Operating Systems – UNIX Internals

Recommended for:

Lectures: 24 h

Graduate students in Computer Science and Computer Systems

The course last ran:

New course. An operating system course last ran in 1992.

Goals:

To give detailed knowledge about theory and practice of operating systems. UNIX will be used as an example throughout the course.

Prerequisites:

Basic knowledge about operating systems, corresponding to the undergraduate courses TDDA21 Concurrent Programming (Processprogrammering) or TDDB01 System Software (Systemprogramvara).

Organization:

Lectures

Contents:

History, Process Management, Memory Management, File Systems, I/O and Device Drivers, Interprocess Communication, Multiprocessing and Threads, STREAMS.

Literature:

Uresh Vahalia: "Unix Internals, the New Frontiers", Prentice Hall, 1995.

Teachers:

Lars Viklund

Examiner:

Peter Fritzson

Schedule:

Spring 1997

Examination:

Written exam + optional implementation project

Credit:

3+1 p

Recommended Master Courses

C3-courses

TDDA14	**AI Programming**
TDDB12	**Concurrent Programming**
TDDA37	**Compiler Construction**
TDDB38	**Database Technology**
TDDA41	**Logic Programming**
TDDA43	**Programming Theory**
TDDA99	**Psychology of Communication**
TDDB42	**Semantics of Programming Languages**
TDDB60	**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**
TDDB10	**Human-Computer Interaction**
TDDB15	**Computer Aided Software Engineering for Development and Maintenance**
TDTS41	**Computer Networks**
TDTS51	**Advanced Computer Architecture**

SVP-courses

HIIC 63	**CASE-verktyg i systemutveckling**
HIIC 62	**Datorteknik och datornät**
HIIC 69	**Människa-Datorinteraktion**
HIIC 66	**Objektorienterad systemutvekling**
HIIC 61	**Prototyping, systemutvecklingsmetoder och verktyg**
HIIC 65	**Samhällsvetenskaplig kunskapsbildning**

Laboratory-oriented Courses and Activities

Like the graduate courses laboratory-oriented Courses and Activities are open for all graduate 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.

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.**

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

Laboratory-oriented Courses and Activities

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.

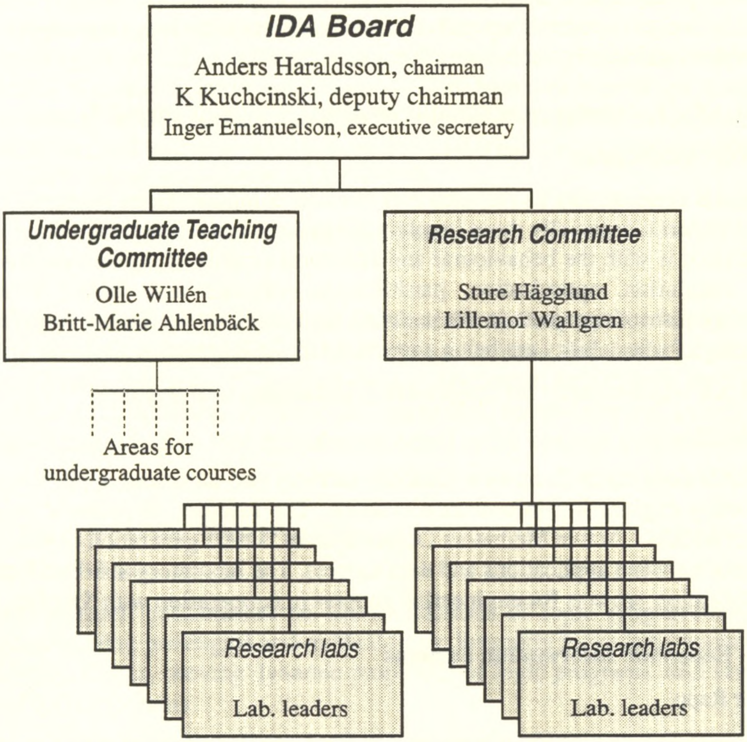
The Laboratory for Temporal-System Correctness and Algorithms (TOSCA)

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


IDA Board
Anders Haraldsson, chairman
K Kuchcinski, deputy chairman
Inger Emanuelson, executive secretary
Undergraduate Teaching Committee
Olle Willén
Britt-Marie Ahlenbäck
Research Committee
Sture Hägglund
Lillemor Wallgren
Areas for undergraduate courses
Research labs
Lab. leaders
Research labs
Lab. leaders

The strategic and executive responsibilities for research projects and graduate education

Ashort presentation of the sixteen research laboratories at the department:

ACTLAB – Laboratory for Complexity of Algorithms

Per-Olof Fjällström

ACTLAB is concerned with the design and analysis of efficient algorithms (sequential and parallel) and data structures for combinatorial and geometric problems arising in computer science and the study of the inherent complexity of these problems in simple models of computation. One application area, studied in the context of CENIIT, is efficient algorithms for three-dimensional geometrical problems.

ASLAB – Application Systems Laboratory

Sture Hägglund

ASLAB is oriented towards the study of knowledge-intensive approaches to software development, including aspects of human-computer interaction. There are currently four subgroups dealing with cognition technology (Hägglund), knowledge engineering (Eriksson), software engineering (Wohlin) and usability matters (Löwgren). Special areas of interest are software quality and reliability, software process improvement, usability engineering and user interface design support, cooperative expert systems, knowledge management and knowledge acquisition, and intelligent tutoring systems.

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, especially those involving very large-scale integrated circuits (VLSI). The major concern is with the behavioural and structural aspects of digital systems specification, design, simulation, optimization, partitioning, synthesis and formal verification methods. Currently research projects in the areas of high-level synthesis, hardware/software codesign, and design for testability are being carried out in the laboratory.

EDSLAB – Laboratory for Engineering Databases and Systems

Tore Risch

EDSLAB does research on new database services and advanced applications in particular for supporting engineering information systems. The research is centered around the umbrella project AMOS - Active Mediators Object System, a next generation object-relational database mediator between applications and data sources. The laboratory also conducts database-oriented application projects for engineering support in cooperation with Swedish industry.

EIS – Economic Information Systems

Birger Rapp

EIS covers communication of information from people to/from systems or between people and the design of information systems supporting this communication. Research projects concern information support, agency theory, IT and organizational solutions, computer simulation for management training and decision support, business control and accounting and auditing.

IISLAB – Laboratory for Intelligent Information Systems

Nahid Shahmehri

IISLAB studies theory and methods for advanced information systems, including object-orientation, computer supported cooperative work, process modelling, workflow management, information retrieval and hypertext, and description logics. In a major implementation project a multi-user information system has been developed which supports parallel development of objects, historical information and automatic maintenance of the database via editing of structured objects.

KPLAB - The 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 computer-based reasoning about action and change in the context of cognitive robotics. We are concerned with the development of a uniform theoretical and implementational framkework for specifying and developing agents that perceive, reason and act in changing and incompletely known environments. Related topics of interest are temporal and nonmonotonic reasoning and implementation techniques for such reasoning processes. Currently, we are placing emphasis on the specification and implementation of agent-orriented software systems for use in distributed, hererogenous information landscapes, such as the Inter- and Intranets. Other topics of interest are visual inference, innovative learning technologies and computer-supported education, intelligent interfaces, and the development of software agents used for gathering, processing and filtering hypertext and hypermedia based information on the WWW.

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 – Laboratory for Logic Programming

Jan Maluszynski and Ulf Nilsson

LOGPRO long term research concentrates on the foundations of logic programming and relations to other programming paradigms and methodology. Presently focus is on query-optimization of deductive databases, proving dynamic properties of logic programs, and verification and synthesis of logic programs.

MDA – People, Computers and Work

Toomas Timpka

MDA conducts research into *information system development and use in working life* from the points of view of computer science, psychology, and social organization of work development. Within the MDA-group, activities at the Department of Computer and Information Science and the Medical Faculty have been coordinated to develop and evaluate experimental information systems.

NLPLAB – Natural Language Processing Laboratory

Lars Ahrenberg

NLPLAB is engaged in research on theoretical and applied natural-language processing. The theoretical research is concerned with linguistic knowledge representation from the syntactic level to the discourse level and methods for robust processing of natural langauge. The applied research covers computer-aided translation and document generation, dialogue systems and help systems.

PELAB – Programming Environments Laboratory

Peter Fritzson

PELAB conducts research in the area of tools and programming languages for software development and maintenance. Current projects include tools for semi-automatic bug location, debuggers for parallel languages, dependence analysis of programs, generation of efficient compilers from denotational semantic specifications, very high level languages and programming environments for scientific computing, and generation of parallel code for mathematical models.

RTSLAB - Real-Time Systems Laboratory

Anders Törne

RTSLAB research covers tools, methods and architectures for the design of software intensive real-time systems. In particular this includes timing analysis and design synthesis tools based on discrete modelling methods, database modelling of activities for control and simulation, and stratified real-time software architectures. The laboratory also conducts applicative research and case studies in cooperation with industry, for example in the automation and control area.

TASLAB - The Autonomous Systems Laboratory

Dimiter Driankov

TASLAB's research is aimed at developing the theoretical basis for the design and analysis of systems with high degree of autonomy. Such systems have two main characteristics: high performance, i.e., the ability to perform well under significant uncertainties in their environment for extended periods of time, and unassisted action, i.e., the ability to compensate for significant system failures without external intervention. 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, the design and analysis of discrete event control systems, planning and replanning in sequential control, and software aspects for layered architecture autonomous systems.

TOSCA - The Laboratory for Temporal-System Correctness and Algorithms

Christer Bäckström

TOSCA conducts research on algorithmic and complexity issues in action planning and temporal reasoning; modelling and verification of hybrid systems; and model-based diagnosis of dynamical systems. TOSCA has many cross-disciplinary research connection with other departments and research groups, both on theoretical and applied topics.

VITS - Development of Information Systems and Work Contexts

Göran Goldkuhl

VITS is a research group studying information systems development in relation to organisational aspects. Special research interest/projects on: Methods for change analysis, information requirements analysis and informations systems evaluation. Strategies for information systems architecture. Relations between methods and CASE tools (CASE shells).

Faculty engaged in the graduate study programme

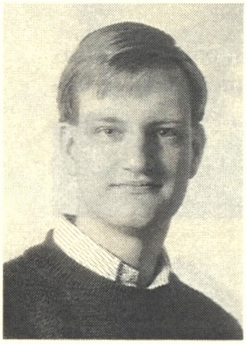
Lars Ahrenberg, Ph. D., Uppsala 1987. Professor of computational linguistics. Group leader, NLPLAB.

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

Sten Andler,

Sten F. Andler, Ph. D., Carnegie-Mellon University 1979.
Professor of Computer Science (*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.



Christer Bäckström, Ph. D., Linköping 1992. Associate professor *(docent, universitetslektor)*. Group leader, TO-SCA-LAB.

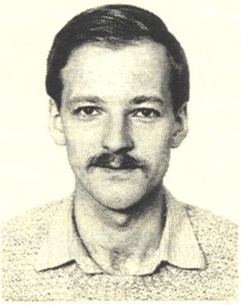
Planning and temporal reasoning, algorithms and complexity, model-based diagnosis.

Nils Dahlbäck, Ph. D., Linköping 1992. Assistant professor, (*universitetslektor*) in 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, universitetslekto*r), 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. 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. Assistant 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, California 1991–94.

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. Professor *(adjungerad professor)* of Economics of Information Technology. Several previous affiliations.

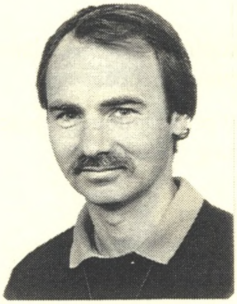
Management of information technology.

Per-Olof Fjällström, Ph. D., Stockholm 1985. Associate professor *(universitetslektor)*, theoretical computer science. Previous affiliation KTH and IBM. Group leader, ACTLAB. Director of graduate study programme.

Computational geometry, analysis of algorithms, data structures.

Peter Fritzson, Ph. D., Linköping 1984. Professor of Programming Systems and Software Engineering. Group leader, PELAB. Previous affiliations Sun MicroSystems, USA, 1985-86, Linköpings Univ. Hospital 1976-79, LiU Physics dep., 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. Professor of informatics, Jönköping International Business School. Information systems development. 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. Department chairman. Previous affiliation Uppsala.

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

Mary Helander, Ph. D., SUNY Buffalo 1992. Assistant professor *(universitetslektor)*, ASLAB. Research associate *(forskarassistent)*, IKP Kvalitetsteknik. Previous affiliations, Northeastern University, Boston, IBM Corporation.

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

Roland Hjerppe, lecturer (*universitetsadjunkt*). Group leader, LIBLAB. Previous affiliation 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. Research Committee chairman. Group leader, ASLAB. Previous affiliation Uppsala.

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

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

Artificial intelligence, Natural language processing, especially empirically based computational models of human-computer dialogues.

Mariam Kamkar, Ph. D., Linköping 1993. Assistant professor *(universitetslektor),* computer science. Director of undergraduate studies in System Architecture.

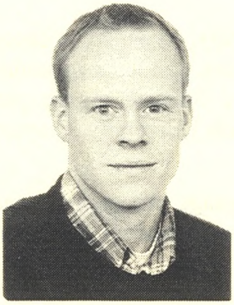
Software maintenance, software testing and analysis, program debugging, program analysis, optimization in compilers, multiparadigm programming languages.

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

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

Bengt Lennartsson, Ph. D., Göteborg 1974. Associate professor (*universitetslektor*), software engineering. Group leader, PELAB 1981-88 and department chairman 1983-90. Previous affiliation for Carlstedt Elektronik AB, Göteborg 1992-94.

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



Jonas Löwgren, Ph. D., Linköping 1991. Associate professor *(docent, universitetslektor*), of human-computer interaction.

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

Jacek Malec, Ph. D., Wroclaw 1987. Research associate (*forskarassistent*). 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.

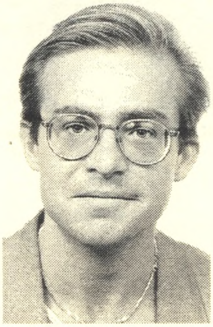
Logic programming, formal language theory, integration of programming paradigms.

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. Associate professor (*docent*) of 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. Group leader, LOGPRO. Previous affiliation State University of New York at StonyBrook, USA.

Logic programming and deductive databases; Evaluation strategies for query processing; program transformation and abstract interpretation.

Nils-Göran Olve, Econ. Dr., Stockholm 1977. Professor *(adjungerad professor)* of management control. Previous academic appointments with Stockholm School of Economics, EIASM (Brussels), and SISU. Since 1986 a senior partner in CEPRO, a management consultancey in Stockholm.

Strategic and economic issues in connection with IT and information. Management control systems to support modern modes of organisation and business activities.

Zebo Peng, Ph. D., Linköping 1987. Associate professor *(docent, universitetslektor*), computer systems.

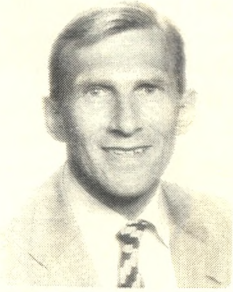
Automated synthesis of digital systems, formal description of hardware, hardware/software co-design, design for testability, VLSI, computer architecture.

Birger Rapp, Econ. Dr., Stockholm 1974. Professor of economic information systems. 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 data bases, active databases, and real-time databases.



Rolf Rundfelt, Ph. D., Stockholm 1974. Professor *(adjungerad professor)* of economic informations 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 actions and change, cognitive robotics, autonomous agents.

Nahid Shahmehri, Ph. D., Linköping 1991. Assistant professor (*universitetslektor*), computer science. Group leader, IISLAB.

Information management, workflow management, information security, CSCW, programming theory, programming languages, debugging tools, compiling technology.

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

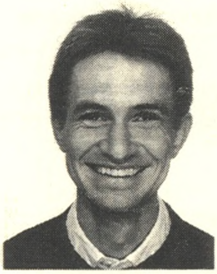
Geographical Information Systems - GIS. Communication of complex data and linking multi disciplinary models in GIS. Research in environmental programs, programs for medical geography, physical planning and decision support.

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 Medical 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. Corporate Research, Västerås. CENIIT manager from 1992.

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



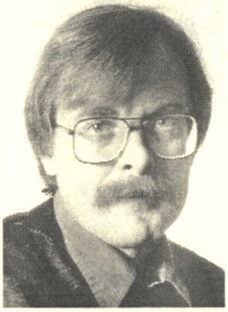
Claes Wohlin, Ph. D., Lund 1991. Acting professor *(docent, tf professor)* of Software Engineering. Previous affiliations Lund University and 5 years of industrial experience in software engineering in 1988-93.

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

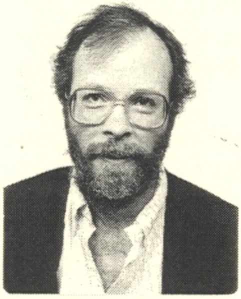
Staffan Bonnier, Ph. D., Linköping 1992. Research associate *(forskarassistent)*. Previous affiliation for Carlstedt Elektronik AB, Göteborg 1993-1994.

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



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.

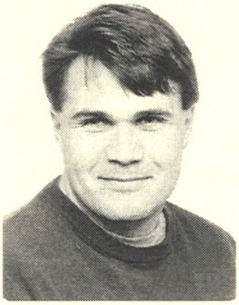


James M. Nyce, Ph. D., Brown 1987. Guest professor, computer and information science. Previous affiliation Brown.

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

Lin Padgham, Ph. D., Linköping 1989. Associate professor *(docent, universitetslektor)*, computer science. Previous affiliation Univ. of Oregon, USA, and Tektronix. Group leader, IISLAB. On leave 1993-95 to the University of Melbourne, Australia.

Inheritance, default reasoning, taxonomical reasoning, object-oriented systems.



Kristian Sandahl, Ph. D., Linköping 1992. Assistant professor *(universitetslektor)*, computer science.

Knowledge Management, Knowledge Engineering, Industrial Software Engineering, Quality Improvement Paradigm, Experimental Research Methods. On leave 1996 to Ericsson Radio Systems, Linköping.

Barbara Wasson, Ph.D. Saskatchewan 1990. Professor of pedagogical information systems (University of Bergen). Previous Affiliations Applied Cognitive Science (University of Toronto), Norwegian Telecom Research, NORUT Information Technology. Visiting professor 1994-1997.

Artificial intelligence in education and training, instructional planning, collaberative learning and intelligent learning environments.

E
LINKÖPINGS UNIVERSITET
LINKÖPINGS UNIVERSITET
-91

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