DEPARTMENT OF COMPUTER AND INFORMATION SCIENCE (IDA)

INTERNATIONAL GRADUATE SCHOOL IN COMPUTER SCIENCE

COURSES 95/96

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 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 primary aim of educating competent researchers at this high-quality establishment. Methods of continually assessing progress and results and proposing improvements to achieve this end are considered essential.

- 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 14 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.

- 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

Graduate Courses Spring 1996

Course	Course literature	Author	Jan	Febr	March	April	May	June	Day	Time	Lokaler
Complexity Theory Credit: 4p Per-Olof Fjällström	1. Introduction to the theory of Complexity	1. D.P.Bouvet P. Crescenzi		This is an undergraduate course - separate schedule							
Computer Algebra and Object Oriented Mathematical Modeling Credit: 3+1p Peter Fritzson							Start on May 7		Tues	9-12	Elogen
Computer Network Services and Protocols Credit: 3p Peter Fritzson	1. TCP/IP Illustrated, Volume1: The Protocols	1. W. Richard Stevens			Start on March 5				Tues	9-12	Estraden
Distribuated Objects Credit: 3+1p Johan Fagerström							Start on May 6		Mon + Wed June 5, 12	9-12 9-12	Estraden 20/5 Belöningen
Ekonomisk styrning Credit: 5p Nils-Göran Olve		1. Johansson-Östman					Vecka 22	Vecka 25, 26			
Topics in Software Engineering Credit: 3 p Bengt Lennartsson	1.Software Engineering	1. Ian Sommerville	Start on Jan 15						Mon Tues	14-17 9-12	
Information Retrieval and Information Filtering Credit: 3p Nahid Shahmehri	1.Introduction to Modern Information Retrieval	1. Salton, Gerard; McGill, Michael J.			Start on March 6				Wed	10-12	Eliten
Introduction to Description Logics Credit: 4+1p Nahid Shahmehri						Start on March 18			Mon	13-16	Elogen
IT-ekonomi och informationsekonomi Credit: 3p B. Rapp, T. Falk, N-G. Olve	1. Investing in Information Technology: Managing the Decision-making Process	1. Geoff Hogbin, David V. Thomas	Start on Jan 18								
Mathematical Systems Theory Credit: 3p J Malec	1. Abstract Systems Theory.	1. Mesarovic, Takahara				Start on April 25			Thur	9-12	Elogen

Graduate Courses Spring 1996

Course	Course literature	Author	Jan	Febr	March	April	May	June	Day	Time	Lokaler
Methodology of Research in Computer Science Credit: 3p (+3p) Sture Hägglund			Jan 22	Febr 5, 12					Mon	13-15	
Modeling and Control of Logical Discrete Event Systems Credit: 2p - 4p Staffan Bonnier	1. Modeling and Control of Logical Discrete Event Systems	1. Kumar, Garg		Start on Febr 9					Fri	13-15	Eliten
Modellering och värdering av systemutvecklingsmetoder Credit: 5p Göran Goldkuhl, Anders G. Nilsson	1.Understanding and evaluating methodologies - NIMSAD	1. Jayaratna			Separate schedule - please turn over						
Network Database Design Credit: 4+2p Tore Risch	1.Transaction Processing, Concepts and Architectures 2. A Tuple Oriented Recovery Method for Continuously Available	1. Gray 2. Hvasshovd			Start on March 1 - May 24				Every Second Fri	9-12 13-15	Apr 12, May 26 Belöningen April 26 ISY konf rum May 24 Estraden
Real-Time Systems Credit:3p (+2) Anders Törne	1.Advances in Real-Time Systems	1. Sang H.Son			Start on Febr 29				Thur	9-12	Elogen
Utredningsmetotik och Kvantitativa metoder Credit: 10p Birger Rapp					Separate schedule						

Modellering och värdering av systemutvecklingsmetoder

Wednesday	Jan 24	15-18	Eliten, Linköping
Monday	Feb 26	9-13	Jönköping
Monday	Mar 11	9-13	Jönköping
Monday	Mar 25	9-13	Jönköping
Thursday	Apr 11	9-12, 13-15	Betinget, Linköping
Thursday	Apr 25	9-12, 13-17	Betinget, Linköping
Wednesday	May 8	13-16	Estraden, Linköping
Wednesday	May 29	9-12, 13-17	Estraden, Linköping
Thursday	Apr 4	10-12	R19 (C-huset)
Tuesday	Apr 23	15-17	FO4
Thursday	Apr 25	10-12	FO7
Wednesday	May 8	13-15	FO7
Friday	May 10	13-15	FO5
Tuesday	May 14	15-17	FO5
Wednesday	May 22	8-10	FO6
Friday	May 24	13-15	FO6

Complexity Theory

Thursday	Mar 21	8-10	FO5
Friday	Mar 22	10-12	FO7
Monday	Mar 25	13-15	FO7
Wednesday	Mar 27	15-17	FO4
Friday	Mar 29	10-12	FO6
Tuesday	Apr 2	8-10	FO7

Contents

Graduate School of Computer and Information Science Graduate Program 1995-1996..........1

General Graduate Courses Fall 1995:

Advanced Computer-based Learning Environments, 3p+1p, B Wasson..................3
Activity Theory: Applications in Computer Science, 5p, T Timpka 4
Applied Rewriting, 3p, J Maluszynski, U Nilsson 5
Aspects of Scientific Writing, 3p, U Nilsson...6
Avhandlingsanalys-Informationssystem, 2-3p, A.G Nilsson 7
Cognitive Semantics, 4+1p, R Hirsch 8
Configuration Management, 3p, P Fritzson, B Magnusson 9
Formal specification of reactive systems: theory and practice, 3p,
S Nadjm-Tehrani, J Malec...10
Information Retrieval 1: Traditional and newer approaches, 3p, R Hjerppe.........11
Introduction to Research Methodology in Computer Science, 2p, S Hägglund......12
Kunskapsutveckling-teori, projektering, genomförande, 3p (del 1) + 3p (del 2)
Göran Goldkuhl..13
Kvalitativ analys och teoriutveckling, 3p, G Goldkuhl... 15
Logical- and Performance Debugging of Parallel Programs, 4p, O Hansen,
P Fritzson 16
Management of Information Technology, 2,5+2,5p, T Falk..................................... 19
Principles of Modern Database Systems, 4+2p, T Risch...20
Topics in Software Engineering, 3p, B Lennartsson 21

General Graduate Courses Spring 1996:

Complexity Theory, 4p, P-O Fjällström..23
Computer Algebra and Object Oriented Math. Modeling, 3+1p, P Fritzson......... 24
Computer Network Services and Protocols, 5p, P Fritzson 25
Distributed Objects, 3+1p, J Fagerström..27
Ekonomisk styrning, 5p, N-G Olve..28
Evaluative Methodology, 3+2p, B Lennartsson...29
Information Retrieval and Information Filtering, 3p, N Shahmehri 31
Introduction to Description Logics, 4+1p, N Shahmehri..32
IT-ekonomi och informationsekonomi, 3p, B Rapp, T Falk, N-G Olve 34
Mathematical System Theory, 3p, J Malec..35
Methodology of Research in Computer Science, 3p (+3p) S Hägglund....................37
Modeling and Control of Logical Discrete Event Systems, 2p-4p, S Bonnier,
J Maluszynski...38
Modellering och värdering av systemutvecklingsmetoder, 5p, G Goldkuhl,
A.G Nilsson..39
Network Database Design, 4+2p, T Risch 40
Real-Time Systems, 3p (+2), A Törne...42
Utredningsmetodik och Kvantitativa metoder, 10p, B Rapp 43

Graduate Courses in the School of Eng. 44

Recommended Master Courses 45

Laboratory-oriented Courses and Activities...46

Research Organization and Laboratories 48

Faculty...................................53

Graduate School of Computer and Information Science
Graduate Programme

General Information
about Graduate Studies in Computer and Information Scince

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.

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 secretary of research (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 1995/1996

This program contains the following types of courses:

- **Graduate Courses at the Graduate School of Computer and Information Science**
- **Graduate Courses in the School of Engineering**
- **Recommended Master Courses**
- **Laboratory-Oriented Courses and Activities**

It also includes presentations of

- **Research Organization**
- **Faculty**

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

Introduction meeting for graduate students.

An Introduction meeting will take place on Tuesday, August 29, 1995 at 13.15 in Estraden, E-building 1st floor.

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 and Per-Olof Fjällström, tel. 013-28 24 12, or for a particular course from the person responsible for the course.

Linköping, June 14, 1995

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

Advanced Computer-based Learning Environments

Recommended for: **Lectures:** 24 h
Graduate students in all areas

The course last ran:
New course.

Goal:
To introduce and investigate advanced computer-based learning environments as a research area.

Prerequisites:
No specific prerequisites.

Organization:
A combination of lectures, seminars, and group work

Contents:
Topics such as:
Knowledge-based Education and Training Systems
Intelligent Tutoring Systems; Intelligent Learning Environments; Adaptive Learning Environments; Automated Instructional Design, Intelligent On-line Assistance, ...
Simulation-based Learning Environments
Educational Multimedia
Pedagogical and Psychological Issues (e.g. situated learning, constructivism, motivation...)
Computer Supported Cooperative Learning
Computers as Cognitive Tools
TeleTeaching (e.g. Virtual Classrooms / Campuses; Distance Learning; Internet-based Learning)

Literature:
Articles and book chapters.

Teachers:
Barbara Wasson
guest lecturers

Examiner:
Barbara Wasson

Schedule:
September - December 1995

Examination:
Active seminar participation.
Term paper or programming assignment.

Credit:
3 points + 1

Activity Theory: Applications in Computer and Information Science

Recommended for: **Lectures:** 20 h
Graduate students in applied Computer and Information Science.

The course last ran:
New course

Goals:
To introduce Activity Theory as a framework for study of Information Systems.

Prerequisites:
Registrated as a doctoral student.

Organization:
Lectures, literature studies, group reports, term paper.

Contents:
Basic concepts of Activity Theory; Theoretical foundation; History of Activity Theory; Application in other areas; Applications in Computer Science; Case examples.

Literature:
To be announced.

Teachers:
Yrjö Engeström
Toomas Timpka

Examiner:
Toomas Timpka

Schedule:
October 1995

Examination:
Term paper

Credit:
5 points

Applied Rewriting

Recommended for: **Lectures:** 28h
Graduate students in computer science and systems

The course last ran:
1993/94

Goals:
The objective of the course is to give a uniform view of different rewriting formalisms applied in different context. Such a general theory of rewriting provides a platform for studying and classifying various paradigms of computation.

Prerequisites:
Formal languages

Organization:
Lectures

Contents:
A general concept of rewriting. Computing functions by rewriting (lambda-calculus and combinatory logic). Abstract data types. Computing relations by rewriting. We show that the well-known concept of context-free rewriting (grammar) extends naturally to logic programs, attribute grammars, and two-level grammars. In this way we get a unified perspective of grammars (capable to describe type 0, type 1 and type 2 languages) and programs. Defining semantics of programming languages in terms of rewriting systems. Foundations of integration of functions and relations. Equational unification and various kinds of narrowing in the perspective of integration of logic programming and functional programming.

Literature:
Collection of papers and book chapters.

Teachers:
J Maluszynski & U Nilsson

Examiner:
J Maluszynski

Schedule:
November 1995 - March 1996

Examination:
Homework and active participation

Credit:
3 points

Aspects of Scientific Writing

Recommended for: **Lectures:** ~15h
Everyone

The course last ran:
1994/95

Goals:
To envisage problems and give practical hints on how to improve the writing.
To help speed up the writing process and prepare participants for potential problems.
To increase awareness of the writing process and initiate further discussions.

Prerequisites:
The course is given in Swedish

Organization:
Lectures (50%), group activities (50%)

Contents:
Scientific writing, the writing process, language and content, reviewing, the scientific paper, typography, the dissertation.

Literature:
A. Jarrick and O. Josephsson. *Från tanke till text.* Studentlitteratur 1988.
J. M. Swales and C. B. Feak. *Academic Writing for Graduate Students.* Univ. of Michigan Press, 1994.

Teachers:
Invited lecturers

Examiner:
Ulf Nilsson

Schedule:
Mid-September through October 1995

Examination:
Homework and active participation

Credit:
3 points

Note:
The course will precede the IDA Conference in Computer and Information Science, held in November 1995.

Avhandlingsanalys - Informationssystem

Recommended for: **Lectures:** 16 h
Doktorander i informationssystemutveckling, ekonomiska informationssystem och informatik

The course last ran:
Ny kurs.

Goals:
Öka förmågan att granska godkända lic- eller doktorsavhandlingar inom området informationssystem som lagts fram vid ett svenskt universitet/högskola.

Prerequisites:
Inga särskilda krav.

Organization:
Kursen inleds med att gå igenom lämplig referensram för att tolka och analysera avhandlingar. Sedan genomförs en självständig analys av en egenhändigt utvald avhandling inom informationssystem. Möjlighet finns även att göra en jämförande analys av två avhandlingar.

Contents:
Metoder för att granska och tolka vetenskapliga publikationer. Meta-modellering som verktyg för att beskriva innehållet i avhandlingar. Opponentens roll vid granskning av avhandlingar. Kriterier för att bedöma avhandlingar. Att skriva avhandlingar i olika internationella kulturer. Egenhändig granskning av hela forskningsprocessen bakom en vald avhandling med analys av grundantaganden, forskningsmetodik, resultat och slutsatser.

Literature:
Lic-/doktorsavhandlingar i informationssystem. Artiklar om avhandlingsanalys. Alvesson, M. & Sklödberg, K. (1994). *Tolkning och reflektion,* Studentlitteratur, Lund.

Teachers:
Anders G. Nilsson

Examiner:
Anders G. Nilsson

Schedule:
November - December 1995

Examination:
Skriftlig rapport (uppsats) om egen avhandlingsanalys. Seminarier.

Credit:
2-3 p

Cognitive Semantics

Recommended for: **Lectures:** 36 h

Graduate students

The course last ran:

This is new course

Goals:

Critical introduction to cognitve semantics in relation to fundamental questions in the semantics of natural languages- i.e. the relation between language and cognition, language and communication.

Prerequisites:

Equivalent of Linguistics Basic Course (C-program)

Organization:

Lectures and discussion seminars with active participation from course members.

Contents:

Linguistic categories, Prototype theory, Stereotypes, Concepts, Mental images, Mental models, Mental representation, Cognitive discourse semantics.

Literature:

Umberto Eco et al (1988) *Meaning and Mental Representations*
John Taylor (1989) *Linguistic Categorization Prototypes in Linguistic Theory*
Peter Gärdenfors (1992) *Blotta Tanken*
Journal Articles

Teachers:

Richard Hirsch

Examiner:

Richard Hirsch

Schedule:

September 1995 - Spring 1996

Examination:

Written or oral exam and course paper

Credit:

4 + 1 points

Configuration Management

Recommended for: **Lectures:** 20h

Graduate students in Computer Science

The course last ran:

New course

Goals:

To learn common terminology and organization techniques, experience from using traditional and modern tools. Acquaintance with current research in this area.

Prerequisites:

Programming courses, software engineering theory or practice, experience from SW projects.

Organization:

Lectures + self study + projects + presentations.

Contents:

Introduction and background:
  Traditional terminology, common systems and tools.
  Why configuration management (CM) is a problem.
  Work organization in large organizations, large projects.
Modern CM systems
  Functionality and use.
Current research in CM
  CM for highly structured data such as in CAD/CAM systems
  CM for teams of programmers.

Literature:

Collection of articles.

Teachers:

Boris Magnusson

Examiner:

Peter Fritzson, Boris Magnusson

Schedule:

October - November 1995

Examination:

Presentations + homework

Credit:

3 points

Formal specification of reactive systems: theory and practice

Recommended for: **Lectures:** 24 h
Computer Science and Systems students

The course last ran:
New course

Goals:
To present the theory of reactive system specification based on communicating state machines (e.g. various sorts of transition systems or statecharts) and to give opportunity to gather a hands-on experience of advantages and drawbacks of using a commericial tool (Statement) develοpped for this purpose.

Prerequisites:
Formal Languages and Automata Theory (TDDA 89)

Organization:
A series of 12 meetings: lectures, seminars and an introductory lab.

Contents:
Theory behind the language of Statecharts;
Semantics for variants of Statecharts, Connections to transition systems;
Formal analysis and approaches to verification
Selected commercial tool: Statemate

Literature:
Articles distributed before and during the course.

Teachers:
Simin Nadjm-Tehrani, Jacek Malec

Examiner:
Simin Nadjm-Tehrani, Jacek Malec

Schedule:
October - December 1995

Examination:
Presentation of seminar papers and practical assignments with documentation.

Credit:
3 points

Information Retrieval I: Traditional and newer approaches

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

The course last ran:
1990/91

Goals:
Provide an overview of traditional issues and solutions in information retrieval .

Prerequisites:
None

Organization:
Reading assignment a will be specified or distributed one week before each meeting. Students will take turns in presenting the contents of the reading assignment and in leading the discussions of issues arising from it.

Contents:
In Information Retrieval (IR) some of the main concerns are:

- the design of databases from a descriptive and a technical point of view
- the design and use of vocabularies for description as well as query formulation
- the translation of "documents" and information requests to matchable items
- methods of matching, and file organizations
- interfaces and interaction in searching
- methods for evaluation

Literature:
To be decided later

Teachers:
Roland Hjerppe

Examiner:
Roland Hjerppe

Schedule:
October - December 1995

Examination:
A written report on a chosen topic

Credit:
3 points

Introduction to Research Methodology in Computer Science

Recommended for: **Lectures:** 16 h
New graduate students.

The course last ran:
Fall 1994.

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 1995

Examination:
Written examination and seminar activity.

Credit:
2 points

Kunskapsutveckling - teori, projektering, genomförande

Recommended for: **Lectures:** 24 h

Forskarstuderande inom informationssystemutveckling, samt andra med intresse för kunskapsteori och forskningsmetodik.

The course last ran:
1994/95

Goals:
Ge översikt över kunskapsteoretiska och forskningsmetodologiska grunder för empiriska studier, samt öka förmågan att planera och genomföra sådana studier.

Prerequisites:
Inga särskilda krav.

Organization:
Delvis samläsning med kursen Samhällsvetenskaplig metodik SVL3/ADB3. Detta innebär gemensamma föreläsningar, men examinationen är helt skild från grundkursen. Del 1 av doktorandkursen innebär utförande av kunskapsprojektering för en forskningsuppgift. Frivillig del 2 kan genomföras som mindre empirisk studie.

Contents:
Kursen fokuserar forskningsarbete som kunskapsutveckling och särskilt problemställningar inom samhällsvetenskaplig kunskapsbildning.
Kunskapsteori: Olika grundläggande vetenskapliga begrepp, bl.a. teorier, paradigm, definitioner, förklaringar och hypoteser. Olika kunskapsformer. Olika vertenskapliga skolor.
Kunskapsprojektering: Planering av forskningsarbete. Metodik/arbetsfaser i kunskapsprojektering.
Genomförande av empiriska undersökningar: Olika typer av undersökningar (klassificering, strategier, tillförlitlighet). Olika undersökningsmetoder (främst intervju, observation, källanalys).

Literature:
Föllesdal, Wallöe, Elster: *Argumentationsteori, språk och vetenskaplsfilosofi,* Thales
Lundahl, Skärvad: *Utedningsmetodik för samhällsvetare och ekonomer*, Studentlitteratur
Repstad: *Närhet och distans,* Studentlitteratur
Goldkuhl: *Kunskapande*, kompendium, IDA
Ytterligare artiklar i samhällsvetenskaplig metodik

Teachers:
Göran Goldkuhl

Examiner:
Göran Goldkuhl

Schedule:
September - November 1995

Examination:
Utförande av arbetsuppgift som dokumenteras i skriftlig rapport. Arbetsuppgiften bör i första hand bestå av utförande av en kunskapsprojektering med egen vald inriktning (del 1). Arbetsuppgiften kan fortsättas med genomförande av en mindre empirisk undersökning. Resultat från denna undersökning ska dokumenteras tillsammans med särskild analys av forskningsmetodik och kunskapskaraktärisering (del 2). Deltagande på seminarier.

Credit:
3 p (del 1) + ca 3 p (del 2)

Kvalitativ analys och teoriutveckling

Recommended for: **Lectures:** 24 h

Doktorander med intresse för kvalitativa undersöknings- och analysmetoder

The course last ran:
Ny kurs.

Goals:
Öka kunskap om kvalitativ analys av empiriska data och hur empirgrundande kategorier och teorier kan genereras.

Prerequisites:
Grundläggande kurs i forskningsmetodik. Kursen förutsätter grundläggande kunskaper i kvalitativ undersökningsmetodik.

Organization:
Koncentrerade föreläsningar. Eget arbete med tillämpningsuppgifter. Seminarier.

Contents:
Grounded theory som metodologiskt angreppssätt för kvalitativ analys av olika typer av empiriska data. Principer för generering av kategorier. Kunskapsgrundning (bl.a. välgrundad metodutveckling). Teoribegreppet. Konstruktion av teorier. Kunskapsmodellering (av kategorier och teorier).

Literature:
BG Glaser: Theoretical sensitivity, *The sociology Press,* Mill Valley
AL Strauss: *Qualitative analysis for social scientists,* Cambridge University Press

Teachers:
Göran Goldkuhl

Examiner:
Göran Goldkuhl

Schedule:
September - December 1995.

Examination:
Skriftliga rapporter. Seminarium.

Credit:
3 p

Logical- and Performance Debugging of Parallel Programs

Recommended for: **Lectures:** 30h

Graduate students in Computer Science

The course last ran:

New course. The previous course on debugging 1993 was on automated debugging.

Goals:

The participants should gain a working knowledge of concepts of debugging and performance analysis with special emphasis on software for parallel and distributed architectures.

Prerequisites:

Knowledge and experience of programming:

Organization:

Lectures + presentations + exercises

Contents:

Introduction to program debugging

- Logical Debugging
  - Purpose
  - Requirements
  - Problems
- Performance Debugging
  - Purpose
  - Requirements
  - Problems
- Introduction to realization techniques to be considered
  - Modes of operation
    (On line, off line...)
  - Interacting with the program execution
   -(Program instrumentation, data collection, execution control)
  - User interaction
   -(Data transformation, filtering, data presentation,
  - graphical user interface)

Program interaction (instrumentation)

  - Execution related event triggering
  - Execution states to be considered
  - Global states of execution in parallel systems
  - Source code instrumentation
  - Object code instrumentation
  - Hardware support
- Data related event triggering
  - Data to be considered
  - Source code instrumentation
  - Object code instrumentation
  - Hardware support
- Producing statistical information

  - Data to be produced
  - Instrumentation techniques
  - Hardware support
- Producing hardware related information
  - Data to be produced
  - Requested hardware support
- Execution control
  - Influencing the state of execution
  - Changing program data
- Standardization issues
  - PTRACE with extensions

Data collection and storage
- Controlling data collection and storage
  - On line
  - Off line
  - Event triggered
- Recording of event traces
  - Local trace collection
  - Hardware supported trace collection
  - Trace recording on distributed file systems
- Producing and recording of derived information
  - Accessing statistical information
  - Local data reduction
  - Global data reduction
- Standardization issues
  - SDDF
  - TDL

Data transformation
- Translation between high level languages and machine code
  - Source code lines
  - Data structures
  - Functions and Methods
  - Threads and Processes
  - Program modules
- Data reduction
  - Filters
  - Data base approaches
- Standardization issues
  - Symbol table formats
    - COFF
    - DWARF

User interface
- Levels of user interfaces
  - Tool construction sets
  - Interfaces with strong user guidance

- Design principles for debugging interfaces
  - General principles (e.g. Source code orientation)
  - Support for data parallelism
  - Support for functional parallelism
- Presentation of event related data
- Space-Time diagrams
  - Focussing on relevant information
- Presentation of statistical data
  - Selection of relevant information
  - Data presentation
- Massive parallelism
  - Problems
  - Solutions
    - Browsing and zooming
    - Automated focusing

Examples
- DETOP
- PATOP

Literature
Papers & Manuals

Teachers:
Olav Hansen

Examiner:
Olav Hansen, Peter Fritzson

Schedule:
September - October 1995

Examination:
Written exercises and programming exercises.

Credit:
4 points

Management of Information Technology

The course last ran:
New course.

Organization:
The course is built on lectures, seminars and case studies. There will be individual as well as group assignments on readings. During the course you are expected to prepare for case studies and submit individual written comments.

Contents:
The theme of the course is the role of modern information technology in business and society and its strategic and organisational consequences.

Literature:
Harvard Business Review: *Revolution in Real Time.* Managing InformationTechnology in the 1990's. Boston: Harvard Business School Press, 1991.

Keen, P.G.W.: *Shaping the Future. Business Design Through Information Technology,* Boston: Harvard Business School Press, 1991.

Strassman, P.A.: *The Business value of Computers.* An Executive's Guide. New Canaan, Conn: The Information Economic Press, 1990.

Teachers:
Thomas Falk

Examiner:
Thomas Falk

Schedule:
October and December 1995

Examination:
There will be a final written exam, accounting for 80% of your grade. Your preparation of the case studies will account for the remaining 20% of the grade.

Credit:
2,5 + 2,5 points.

Principles of Modern Database Systems

Recommended for: Graduate students	**Lectures**	32h

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.

Periodicity:
Once per Year.

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 Gibbons
-DBMSS architectures
-Data Models (e.g. relational, OO, functional)
-Object Data Management Concepts
-Query processing (OO and relational)
-Concurrency and Recovery
-Versioning
-Database Performance and Benchmarks
-Deductive Databases
-Active Databases
-Temporal Databases
-Main-memory Databases

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

Teachers:
Tore Risch

Examiner:
Tore Risch

Schedule:
November 1995 - January 1996

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

Credit:
4+2 points

Topics in Software Engineering

Recommended for: **Lectures:** 20h
Graduate students at the department

The course last ran:
New contents

Goals:
To give an overview of concepts and models in Software Engineering, and to present selected relevant activities in the area.

Prerequisites:
None.

Organization:
Individual readings, lectures, guest seminars, and paper presentations.

Contents:
Overview of software engineering.
Software architecture and software reuse.
ndustrial development projects.

Literature:
Basic textbook for overview: Ian Sommerville. *Software Engineering.*

Teachers:
Invited speakers, Peter Fritzson, Bengt Lennartsson.

Examiner:
Bengt Lennartsson

Schedule:
During 1995/96.

Examination:
Term papers(s).

Credit:
3 points.

Complexity Theory

Recommended for: **Lectures:** 30 h

Computer Science students.

The course last ran:
A course with similar contents was given in spring -93

Goals:
The systematic study of computational complexity theory has developed into one of the central and most active research areas of computer science. The aim of this course is to present the most significant results obtained in this area of research.

Prerequisites:
Basic course on design and analysis of algorithms.

Organization:
Two lectures weekly, obligatory homework.

Contents:

1. Elements of computability theory
2. Complexity classes
3. The class P and NP
4. The complexity of optimization problems
5. Beyond NP
6. Space-complexity classes
7. Probabilistic algorithms and complexity classes
8. Interactive proof systems
9. Models of parallel computers and parallel algorithms

Literature:
D.P. Bovet, P. Crescenzi. *Introduction to the theory of Complexity.* Prentice Hall.

Teachers:
Per-Olof Fjällström.

Examiner:
Per-Olof Fjällström.

Schedule:
Spring 1996.

Examination:
Obligatory homework. Written solutions to homework must be submitted and each student must be prepared to orally present her solutions.

Credit:
4 points.

Computer Algebra and Object Oriented Mathematical Modeling

Recommended for: **Lectures:** 22h
Graduate students in Computer Science

The course last ran:
New course. The last graduate computer algebra course ran at least 9 years ago.

Goals:
The participants should gain a working knowledge of concepts in computer algebra systems, how to use these systems, and how to construct object oriented mathematical models.

Prerequisites:
Knowing the basic concepts of object orientation, and having.

Organization:
Lectures + presentations + exercises

Contents:

1. Computer Algebra Languages
   - Abstraction concepts
   - Function and rule invocation
   - Algebraic transformations
   - Systems: Mathematica, Maple, Axiom
2. Object Oriented Mathematical Modeling
   - Modeling styles
   - Interfaces, inheritance & encapsulation
   - Interactions between computer algebra and OO
   - Systems: Dymola, ObjectMath, etc.

Literature:
Articles

Teachers:
Peter Fritzson, invited speakers

Examiner:
Peter Fritzson

Schedule:
Spring 1996

Examination:
Programming, transformation and modeling exercises using available systems.

Credit:
3 + 1 points

Computer Network Services and Protocols

Recommended for: **Lectures:** 36 h

Graduate students in Computer Science and Computer Systems

The course last ran:

New course

Goals:

To give an understanding of principles for design and implementation of network services and the underlying protocols. The principles are exemplifies by in-depth studies of existing Internet protocols.

Prerequisites:

Basic knowledge about computer networks (e.g. from the undergraduate course "Computer networks") is preferable, but not necessary. A working knowledge of the C programming language is necessary for the exercises.

Organization:

Lectures and practical exercises.

Contents:

- Introduction to computer networks. Basic principles and terminology (the OSI model etc.)
- The physical and link layers. Examples: IEEE 802, FDDI, ATM, ISDN.
- Network security.
  The Internet: organization and standards.
- Protocols for the network and transport layers. Examples: IP, UDP, TCP, ICMP, ARP, RARP.
  Programming interfaces to such protocols. Examples: BSD sockets, TLI.
- Point-to-point protocols. Example: SLIP, PPP.
  Simple application protocols. Example: TELNET
  Routing and routing protocols. Examples: OSPF, RIP.
  The domain name service (DNS.)
- Network management tools and protocols. Example: SNMP. (guest lecturer)
- Protocols for E-mail. Examples: SMTP, POP3, IMAP.
  Standards for message encoding. Example: MIME.
  Protocols for other information services. Examples: FTP, NNTP, HTTP.
- Remote procedure calls. Principles, protocols, programming interfaces and tools.
  Examples: Sun RPC and XDR.
- Distributed file systems. Example: NFS
- Authorization. Example: Kerberos
- SUNET: technology, structure, organisation, and history. (Guest lecturer)
- Summary of the course.

Literature:

A textbook on computer networks (e.g. Comer: *Internetworking with TCP/IP*) and protocol specifications (RFCs.)

Teachers:

Lars Viklund, Niclas Andersson, and invited lecturers.

Examiner:
Peter Fritzson

Schedule:
Spring 1996

Examination:
Three individual design and implementation exercises.

Credit:
5 points

Distributed Objects

Recommended for:	**Lectures:**	20h
Graduate students in Computer Science		

The course last ran:
New course

Goals:
The participants shall understand the coming integration of object-oriented and distributed systems. In particular, the participants will understand the importance of standardized middleware used to isolate operating systems, networks and programming languages from object-oriented applications.

Prerequisites:
The participants must understand the object-oriented model and have a general understanding of distributed systems. A working knowledge of C or C++ is required for the extra graduate point project.

Organization:
Lectures + self study + project

Contents:

1. Distributed system architectures
   - variations on client-server
   - traditional ad OO distributed databases
   - distributed objects
2. Integration of systems using distributed objects
3. CORBA
   - The Common Object Model
   - IDL, Interface Definition Language
   - mappings to C, C++ and Smalltalk
   - Object brokering
4. Products
   - IBM SOM/DSOM
   - Digital's ObjectBroker
   - IONA / Expersoft / SUN DOE and other systems
5. Component-based system development
   - OpenDoc
   - OLE 2
6. "The future distributed systems"

Literature:
Handouts of articles

Teachers:
Johan Fagerström

Examiner:
Johan Fagerström

Ekonomisk styrning

Recommended for: **Lectures:** ca 35 h

Forskare med baskunskaper i redovisning, kostnadsintäktsanalys och organisation.

The course last ran:
Hösten 1993.

Goals:
Gemensamt utforska ekonomisk informationspotential som styrmedel internt i organisationer, med särskild betoning på målstyrning och decentralisering.

Prerequisites:
Inga formella.

Organization:
5 heldagars undervisning under IDA/EIS:s forskarveckor.

Contents:
Ekonomisk information som styrmedel i förhållande till organisationens struktur och uppgift. Lönsamhets- och prestationsmätning. Mål: krav på ekonomiska resultat. Planeringsprocesser och internmarknad. Organisation av den ekonomiska styrningen: controllers, stödsystem.

Literature:
Böcker av Johansson-Östman, Samuelson (red) och (prel.) Anthony-Dearden-Govindarajan; artiklar och enstaka kapitel ur andra skrifter, delvis i form av individuella referatuppdrag.

Teachers:
Nils-Göran Olve.

Examiner:
Nils-Göran Olve

Schedule:
Våren 1996.

Examination:
Presentationer och uppsatser under kursens lopp.

Credit:
5 poäng

Evaluative Methodology

Recommended for: **Lectures:** 12h

Students interested in evaluation of new technology.

The course last ran:
New course

Goals:
To give an introduction to qualitative methods for the evaluation of new (software) technology.

Organization:
Lektures, seminars, paper presentations, exercises.

Contents:
Planning the evaluative research.
Methods for data collection; interviews, etc.
Methods for data analysis.
Optional exercise.

Literature:
To be decided.

Teachers:
Invited speakers, Toomas Timpka, Göran Goldkuhl, Bengt Lennartsson.

Examiner:
Bengt Lennartsson

Schedule:
Spring 1996

Examination:
Term papers(s) for 3 points.
A small exercise (planning, collection, analysis, conclusions, report). Additional 2 points.

Credit:
3 + 2 points

Schedule:
Spring 1996

Examination:
Written exam and a programming exercise for extra graduate point(s).

Credit:
3 + 1 points

Information Retrieval and Information Filtering

Recommended for:	**Lectures:**	24 h

Graduate Students. (Orienteringskurs C4)

The course last ran:
New course

Goals:
The course goal is to give an introduction to information filtering techniques and the underlaying technology (statistics, collaboration, natural language understanding, learning). Hands on experience with a few existing systems will be included. The course will include an introduction to the WWW and its relevance and role in information retrieval.
After the course, students should have an understanding of existing information filtering techniques, their limitations and possibilities. They will also have some experience with a few systems.

Prerequisites:
None.

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

Contents:

- Information Retrieval (IR)
- Connection between IR and Information Filtering (IF)
- Fundamentals of IF (Requirements)
- Basic Approaches to IF (IR, collaboration, group reviews, rule based, agent oriented, user modeling, natural language)
- Email and news filtering systems
- Machine learning of user preferences
- Agents: Standalone and Cooperating
- Privacy issues
- WWW including agents for searching the net and WWW as interface to DB

Literature:
Articles

Teachers:
IISLAB members

Examiner:
Nahid Shahmehri

Schedule:
Spring 1996

Examination:
Laboratory Assignments and a Term paper

Credit:
3 points

Introduction to Description Logics

Recommended for: **Lectures:** 30 h

Graduate Students. (Fördjupningskurs C4)

The course last ran:
New course

Goals:
Description logics (or terminological logics or concept languages) are computationally attractive limited subsets of first-order logic. They are used for representing concepts and individuals in a given "world" and then drawing logical inferences from the given information. The inferences center around, but are not limited to, classification. There are a number of implemented description logic systems which are used for building a variety of applications including software management systems, planning systems, telephone configuration systems, and natural language understanding. Description logic systems are particularly suitable for applications where classification is a useful inference. At the conclusion of this course you will have:

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

Prerequisites:
a basic course in logic (e.g. Logic for ida-ites, Logik grundkurs (TDDA15))

Organization:
Seminars (prepared by the participants) + labs

Contents:
- Introduction to the field, what description logics are, KL-ONE
- Representing knowledge in description logics
- The algorithms: classification, subsumption, other inferencing
- Completeness, Correctness, Complexity
- Revision and update of knowledge bases
- Systems: KRIS, LOOM, CLASSIC
- Some applications using description logics
- Connection to databases
- Hot topics: a.o. integrating different kinds of reasoning with description logics (e.g. defaults, part-of, time, actions)

Literature:
Articles

Teachers:
Patrik Lambrix

Examiner:
Nahid Shahmehri

Schedule:
Spring 1996

Examination:
Seminar, lab assignments, summaries

Extra credit for lager project or term paper

Credit:
4+1 points

IT-ekonomi och informationsekonomi

Recommended for:
Ekonomiska informationssystem.

The course last ran:
Första gången.

Goals:
Att förstå sambandet mellan IT och ekonomiska effekter i företag och samhälle

Prerequisites:
Ekonomisk grundexamen.

Organization:
En betydande del av kursen kommer att bestå av tillämpningsövningar där deltagarna får arbeta med exempel på IT-ekonomi och informationsekonomi. Därtill kommer föreläsningar kring metodproblem och genomgångar.

Teachers:
Nils-Göran Olve
Thomas Falk
Birger Rapp

Examiner:
Nils-Göran Olve
Thomas Falk
Birger Rapp

Schedule:
Start våren 1996.

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

Credit:
3 poäng

Mathematical Systems Theory

Recommended for: **Lectures:** 24h

Computer Science and Systems students

The course last run:

1992/93

Goals:

To give an introductory presentation of formal systems theory. To present systems theory as a metalanguage for engineering sciences. To present a generic formal definition and formulation of properties of dynamic systems. To illustrate connections between different application areas of systems approach.

The course in not intended to substitute any control theory or automata theory courses. It is rather intended to serve as a follow-up course, both for computer science students (C-line background) interested in basic control theory concepts, and computer systems students (D-line background) interested in basic theoretical computer science concepts.

Prerequisites:

In principle, no preliminary knowledge except elementary set theory and elementary algebra is required. However, knowledge of the material presented during one or more of the following courses might be helpful.

- Reglerteknik (TSRT 12,13,15,16,18)
- Control Theory (TSRT 35,38)
- Modelling (TSRT 20,50,64)
- Formal Languages and Automata Theory (TDDA 89)

Organization:

A series of 12 lectures, possibly with some classes.

Contents:

1 Definitions and classification
-general system (any relation defined on $\times_{i \in I} V_i$);
-input-output system (any relation defined on $(\times_{i \in I} A_i) \times (\times_{j \in J} B_j)$);
-time system (a relation defined on $A^T \times B^T$);
-dynamic system (a system which can be described using two families of functions (r
-sponse family and state-transition family) fulfilling apropriate criteria);
-classification based on assuming additional structure imposed on inputs, outputs, states, or time.

2 Problems studied within the systems theory framework
-realization and minimal realization problems;
-analysis of systems:
-observability,
-controllability,
-stability;
-composition and decomposition of systems;
-goal-orientedness.

Literature:
M.D. Mesarovic and Y. Takahara *Abstract Systems Theory*/, Lecture Notes in Control and Information Sciences, vol. 116, Springer-Verlag, Berlin, 1989

Alternatively a compendium prepared by myself;

Literature recommended:
L. Padulo and M. A. Arbib *System Theory: a unified state space approach to continuous and discrete systems*/, W.B.Saunders, Philadelphia, 1974
M. D. Mesarovic, D. Macko and Y. Takahara *Theory of Hierarchical, Multilevel Systems*/ Mathemathics in Science and Engineering series, vol. 68, Academic Press, NY, 1970
M. D. Mesarovic and Yasuhiko Takahara *General Systems Theory: Mathematical Foundations*/ Mathemathics in Science and Engineering series, vol. 113, Academic Press, NY, 1975.

Teachers:
Jacek Malec

Examiner:
Jacek Malec

Schedule:
Spring 1996

Examination:
Written assignments (exercises) throughout the course. Term paper.

Credit:
3 points

Methodology of Research in Computer Science

Recommended for:
For graduate students

Lectures: 16 h + Seminars 10-15 h

The course last ran:
1990

Goals:
The goal of this course is to give the student an understanding for methodological problems in the process of performing research in general and in computer science in particular. The course also includes practical issues in connection with selecting a research problem, planning a project, carrying out the research and reporting the results.

Prerequisites:
None

Organization:
The course will have the character of an advanced seminar with presentations on selected topics, mainly by invited lecturers. Some letters will be used for a review of more practical issues in planning and carrying out research. There will aslo be a number of more discussion-oriented seminars, where specific papers or issues are treated. An additional 3 points may be obtained for writing an essay on a topic suggested or approved by the course leader.

Contents:

- Introduction to the theory of science
- Scientific paradigms
- Project conception and planning
- Scientific writing and publication
- Seminars with discussion of papers and dissertations from a methodological point of view

Literature:
To be decided later.
Articles and reports.

Teachers:
Sture Hägglund, Roland Hjerppe, Erik Sandewall, invited lecturers

Examiner:
Sture Hägglund

Schedule:
Spring 1996

Examination:
Active participation in lectures and seminars. A written summary of course material and other completed assignments.

Credit:
3 points (+3 points)

Modeling and Control of Logical Discrete Event Systems

Recommended for: **Lectures:** 20-30 h
Computer Science and systems graduate students.

The course last ran:
New course

Goals:
The field of discrete event systems has emerged to provide a formal treatment of many of the man-made systems such as manufacturing systems, communication networks, automated traffic systems, database management systems, and computer systems that are event-driven. The goal of this course is to present a general theory of *logical* discrete event systems, where the focus is on the *order* of events rather than on their occurrence *times*.

Prerequisites:
Some degree of 'mathematical maturity'.

Organization:
Seminars given by participants + invited speakers.

Contents:

- Introduction to Formal Language and Lattice Theory,
- Control under Complete Observation,
- Control under Partial Observation,
- Control of Non-Terminating Behavior.

Literature:
R. Kumar, V.K. Garg: *Modeling and Control of Logical Discrete Event Systems,* Kluwer 1995.

Teachers:
Staffan Bonnier, Jan Maluszynski

Examiner:
Staffan Bonnier

Schedule
Spring 1996

Examination:
Seminars + homework

Credits:
2p - 4p

Modellering och värdering av systemutvecklingsmetoder

Recommended for: **Lectures:** 40 h

Doktorander i informationssystemutveckling. Av intresse också för doktorander i ekonomiska informationssystem, informatik, datalogi.

The course last ran:

Ny kurs.

Goals:

Öka förståelsen om systemutvecklingsmetoder, och liknande metoder för verksamhetsutveckling, genom modellering och utvärdering. Öka kunskapen om metametoder för modellering och utvärdering av metoder.

Prerequisites:

Inga särskilda krav. Gärna någon grundläggande kurs i systemutveckling.

Organization:

Kursen är uppdelad i två delar: 1) metodmodellering 2) metodvärdering. Koncentrerade föreläsningar. Eget arbete med tillämpningsuppgifter. Seminarier.

Contents:

Metamodellering av metoder för system/verksamhetsutveckling;
Metodanalys/SIMM och andra metametoder för modellering av metoder; olika metodkomponenter i en metametod; utvärdering av metoder; kriterier för metodvärdering; NIMSAD och andra metodansatser för metodvärdering

Literature:

N Jayaratna (1994) *Understanding and evaluating methodologies - NIMSAD*, McGraw-Hill; Artiklar

Teachers:

Göran Goldkuhl (metodmodellering),
Anders G Nilsson (metodvärdering)

Examiner:

Göran Goldkuhl, Anders G Nilsson

Schedule:

Spring 1996

Examination:

Skriftliga rapporter. Seminarium.

Credit:

5 p

Network Database Design

Recommended for:
Graduate students

Lectures: 26 h + 10 h exercises

Periodicity:
Once every second

Goals:
Gain a good understanding of database design especially concerning transaction processing and reliability.

Prerequisites:
Undergraduate courses in computer science. Basic database course.

Organization:
Lectures and seminars covering the core material of the course. Extra points for book project or design project.

Contents:

- Network Database Applications
- Network Database Requirements
- AML, A Modelling Language
- System Model
- Operations of the Database
- Infrastructure Model (HW Model)
- Conceptual Models of Operations
- Logical Node Models
- Service Model
- Abstract Machine Model
- A Reliable Update
- Implementation Model
- Implementation of Concurrency Control
- Implementation of Recovery
- Implementation of Access Structures
- Change Management (Schema upgrades, software change etc.)
- Research topics

Literature:
J. Gray, *Transaction Processing, Concepts and Architectures,* A Reuter Lecture Notes
S-O Hvasshovd, *A Tuple Oriented Recovery Method for Continuosly Available Distributed DBMS on a shared nothing Multi-computer*
Research articles
Modern Database Systems and Distributed Databases preferred.

Teachers:
Mikael Ronström

Examiner:
Tore Risch

Schedule:
Spring 1996

Examination:
Written exam. Design project.

Credit:
4 + 2 points

Real-Time Systems

Recommended for. **Lectures:** 24 h

Computer Science and Systems graduate students.

Periodicity:
Every second year

Goals:
The course will give a survey to different research issues, methods and state of the art in the design of real time systems.

Prerequisites:
Processprogrammering TDDA21 or corresponding prerequisites
Basic knowledge in finite automata, logic and operating systems.

Contents:

- Fault tolerance, atomic actions and exception handling
- Scheduling and Resource Management
- Distributed real time systems
- Specification of real time systems - e.g., RTL, Petri Net Models
- Real Time Languages - synchronous and asynchronous
- RT-system design
- Case study

Literature
Probably from Burns&Wellings: *Real-Time Systems and their programming languages* together with articles and other excepts

Teachers:
Anders Törne

Examiner:
Anders Törne

Schedule:
Spring 1996

Credit:
3p (+ 2p)

Utredningsmetodik och Kvantitativa metoder

Recommended for: **Lectures:** 50-70 h
Alla doktorander.

The course last ran:
Våren och början av hösten 1995.

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 1996.

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

Credit:
10 poäng

Presentationsteknik

Fall 1995, Lars-Gunnar Petersson, IFM, tel 28 12 49

Forskningsetik

Fall 1995, Lars-Gunnar Petersson, IFM, tel 28 12 49

Teknikhistoria

Spring 1996, Lars Strömbäck, IKS, tel 28 18 13

Vetenskapsteori

Spring 1996, Ingemar Nordin, Tema H, tel 28 22 20

Recommended Master Courses

C3-courses

TDDA14	**AI Programming**
TDDA21	**Concurrent Programming**
TDDA37	**Compiler Construction**
TDDA38	**Database Technology**
TDDA41	**Logic Programming**
TDDA43	**Programming Theory**
TDDA99	**Psychology of Communication**
TDDB40	**Rewriting Systems**
TDDB42	**Semantics of Programming Languages**
TDDB45	**Computability and Complexity Theory**
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**
TDDA30	**Programming Theory II**
TDDA32	**Design and Analysis of Algorithms**
TDDB33	**Object-Oriented System Development**
TDDA66	**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**

SVL-courses

CASE-verktyg i systemutveckling
Datorteknik och datornät
Informationssystem och MPS
Människa-Datorinteraktion
Objektorienterad systemutvekling
Programmering i C och principer för programspråk
Prototyping, systemutvecklingsmetoder och verktyg
Samhällsvetenskaplig kunskapsbildning

Laboratory-oriented Courses and Activities

Like the graduate courses these 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. Additional course activities may be announced during the year.

The Laboratory for Complexity of Algorithms (ACTLAB)

Fixed time for lab meetings: **Wednesdays 13-15.**

The Laboratory for Application Systems (ASLAB)

The course *Knowledge Acquisition,* will be given by K Sandahl during spring 1996.
Fixed time for lab meetings: **Thursdays 13-15.**

The Laboratory for Computer Aided-Design of Digital Systems (CADLAB)

The course *Digital ASIC Testing and Testable Design,* will be given by Z Peng and K Kuchcinski during spring 1996 i.e. Tuesdays 10-12, weeks 6-15.
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 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.**

The Laboratory for Programming Environments (PELAB)

Fixed time for lab meetings: **Thursdays 15-17.**

Laboratory-oriented Courses and Activities

The Laboratory for Representation of Knowledge in Logic (RKLLAB)

Fixed time for lab meetings: **Wednesdays 13-15.**

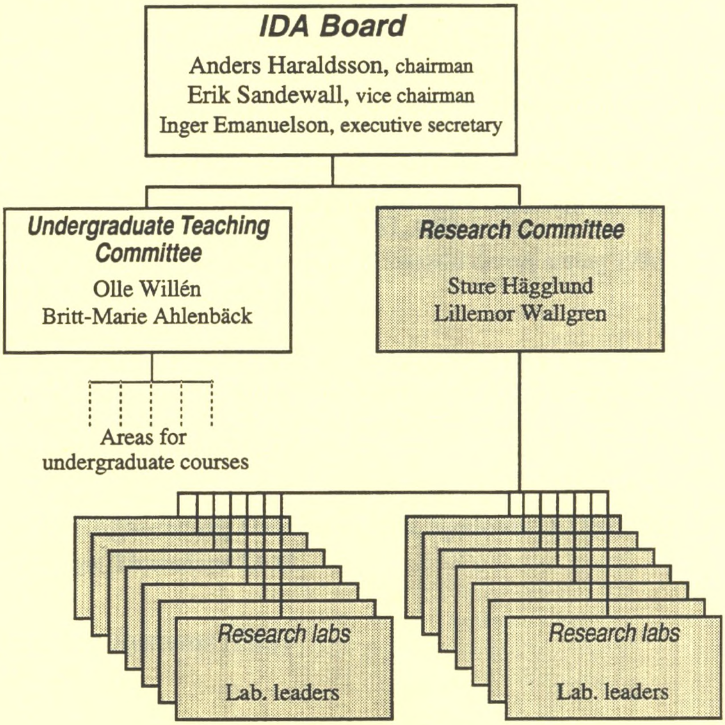
The Real-Time Systems Laboratory (RTSLAB)

Fixed time for lab meetings: **Wednesdays 13-15.**

Information Systems and Work Contexts (VITS)

Fixed time for lab meetings: **Some Mondays 10-12.**

Research Organization and Laboratories


IDA Board
Anders Haraldsson, chairman
Erik Sandewall, vice 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 fourteen 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 three subgroups dealing with cognition technology (Hägglund), knowledge engineering and software engineering (Sandahl) and usability matters (Löwgren). Special areas of interest are industrial software quality and reliability, knowledge management, usability engineering and user interface design support, expert critiquing systems, knowledge acquisition, diagrammatic reasoning 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..

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 algoritms for parsing, generation and unification. The applied research covers natural language dialogue systems and tools for machine-aided translation and document generation.

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.

RKLLAB – Laboratory for Representation of Knowledge in Logic

Erik Sandewall

RKLLAB conducts research on logic-based principles for the design of intelligent autonomous agents. This includes research on non-monotonic logics, logics for reasoning about action and change, fuzzy logic, algorithms for planning and temporal prediction, and related topics. It also includes research on methods for the systematic description of physical systems on a discrete level, and on architectures and tools for complex real-time systems.

RTSLAB - Real-Time Systems Laboratory

Anders Törne

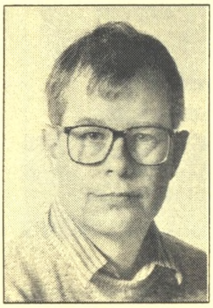
RTSLAB research covers tools, methods and architectures for the design of software intensive real-time systems. In particular this includes timing analysis and design synthesis tools based on discrete modelling methods, database modelling of activities for control and simulation, and stratified real-time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

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 Graduate Study Program



Lars Ahrenberg, Ph. D., Uppsala 1987. Professor of computational linguistics. Group leader, NLPLAB. On partial leave 1994-95.

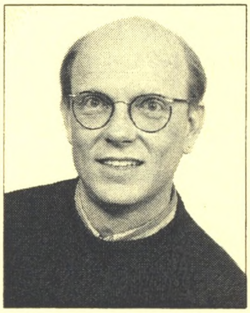
Syntax, semantics and pragmatics of natural language; natural language understanding; natural language interfaces; machine-aided translation.

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.

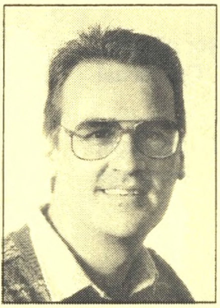
Christer Bäckström, Ph. D., Linköpig 1992. Assistant professor *(universitetslektor)*.

Planning and temporal reasoning, algorithms and complexity for AI problems, representation and reasoning about knowledge.



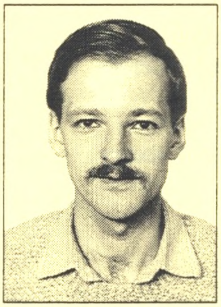
Nils Dahlbäck, Ph. D., Linköping 1992. Acting professor, (*tf professor*) cognitive science. Director of undergraduate studies in cognitive sciences.

Natural language processing, especially empirically based computational models of discourse. Cognitive aspects of discourse coherence in man and machine. Philosophy of mind and its consequences for empirical theories in cognitive science.



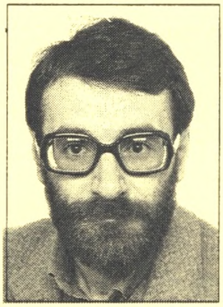
Patrick Doherty, Ph. D., Linköping 1991. Assistant professor (*universitetslektor*), logic and theoretical computer science.

Logical approaches to knowledge representation; reasoning with incomplete information, non-monotonic reasoning, reasoning about action and change.



Wlodek Drabent, Ph. D., Warsaw 1985. Acting professor (*docent, tf professor*), programming theory. On leave from Institute of Computer Science, Polish Academy of Sciences.

Logic programming, programming languages semantics.



Dimiter Driankov, Ph. D., Linköping 1989. Assistant professor *(universitetslektor)*, logic and AI.

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

Henrik Eriksson, Ph. D., Linköping 1991. Assistant professor (*vik. universitetslektor*), informatics. Previous affiliation Stanford University, Stanford, California 1991–94.

Knowledge-based systems, knowledge acquisition, sofwardevelopment environments, software reuse.

Johan Fagerström, Ph. D., Linköping 1988. Assistant professor (*universitetslektor*), computer science. Director of undergraduate studies in System Architecture.

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

Thomas Falk, Econ. Dr., Stockholm 1976. Adjunct professor in economics of information technology. Chalmers University of Technology. Several previous affiliations.

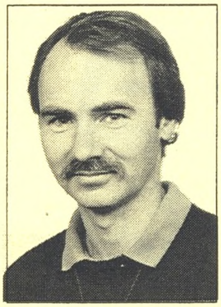
Management of information technology.

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.

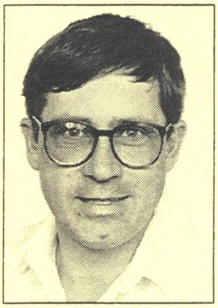
Peter Fritzson, Ph. D., Linköping 1984. Professor of Programming Systems and Software Engineering. Group leader, PELAB. Previous affiliations Sun MicroSystems, USA, 1985-86, Linköpings Univ. Hospital 1976-79, LiU Physics dep., 1975-76.

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



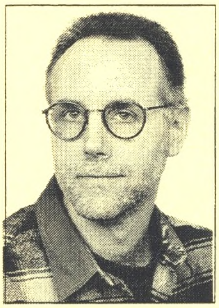
Göran Goldkuhl, Ph. D., Stockholm 1980. Associate professor (*docent, universitetslektor*), administrative data processing. Group leader, VITS. Research director at Centre for studies on Man, Technology and Organization. Previous affiliations Stockholm, Göteborg.

Theories/methods on problem formulation, business process and activity analysis, IS design and evaluation; ISD methods and customization of CASE tools; Humanistic science traditions and qualitative research methods.



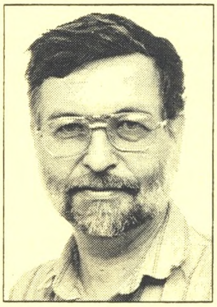
Anders Haraldsson, Ph. D., Linköping 1977. Associate professor (*bitr professor*), computer science. Department chairman. Previous affiliation Uppsala.

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



Richard Hirsch, Ph. D., Gothenburg 1989. Associate professor *(docent, universitetsadjunkt)* of general and computational linguistics. Thesis supervision in the laboratory for natural language processing. Previous affiliation Gothenburg.

Syntax, semantics, and pragmatics of natural languages; discourse analysis; argumentation theory; semiotics; philosophy of language.

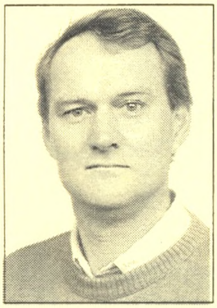


Roland Hjerppe, Universitetsadjunkt. Group leader, LIBLAB. Previous affiliation KTH, DFI and expert mission Tanzania. Visiting Distinguished Scholar at Office of Research, OCLC Inc. in Columbus, Ohio, 1988-89.

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.



Arne Jönsson, Ph. D., Linköping 1993. Assistant professor (*universitetslektor),* computer science. Director of undergraduate studies in cognitive sciences. On leave 1994-95 to Monash University, Clayton, Australia.

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.

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 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 issues in information systems development and software engineering, usability-oriented design support.

Jacek Malec, Ph. D., Wroclaw 1987. Research associate (*forskarassistent*). Previous affiliation Technical Univ. of Wroclaw, Poland.

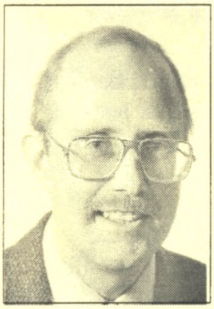
Artificial intelligence, knowledge representation, reactive systems, autonomous systems, system theory.

Jan Maluszynski, Ph. D., Warsaw 1973. Professor of programming theory. Several previous affiliations. Group leader, LOGPRO. On leave 1994-95 INRIA, France.

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

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

Modelling and verification of embedded systems, temporal logic, real-time systems, logic programming.



Anders G. Nilsson, Econ. Dr., Stockholm 1991. Affiliated 94/95. Position at the Institute for Business Process Development (Institute V) and Stockholm School of Economics. Previous affiliations Royal Institute of Technology and Stockholm University.

Business modelling, strategy planning, activity based development, information systems development, maintenance management, application packages, information management.

Ulf Nilsson, Ph. D., Linköping 1992. Assistant professor (*universitetslektor*), computer science. Group leader, LOG-PRO. 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, Adjunct professor in economics of information technology at Stockholm School of Economics (HHS) and Swedish Institute of Management (IFL) from 1981. Visiting professor at EIASM in Brussels 1984-85. Acting professor in Linköping 1993. Since 1986 a senior partner in CEPRO, a management consultancy in Stockholm. Part-time with Swedish Institute of Systems Development (SISU) 1993-95.

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.

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.

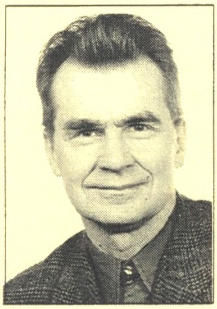
Birger Rapp, Econ. Dr., Stockholm 1974. Professor of economic information systems. Vice president at large for IFORS. Editorial (advisory) boards to EJOR, IJMSD, JORBEL and Omega. President of the Pronova Research and Development Board in Sweden.

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.

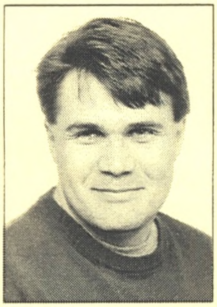
Database support for engineering and scientific applications, e.g., object-oriented databases, heterogeneous data bases, active databases, and real-time databases.

Rolf Rundfelt, Ph. D., Stockholm 1974. Adjunct professor in economic informations systems, especially in Swedish and International External 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. Vice-Chairman of the department. Research Committee Chairman (1983-1990). Group leader, RKLLAB. Several previous affiliations.

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



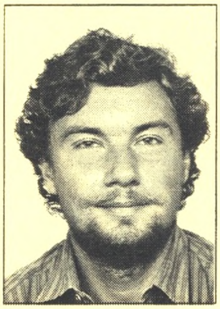
Kristian Sandahl, Ph. D., Linköping 1992. Assistant professor (*universitetslektor*), computer science.

Knowledge Management, Knowledge Engineering, Industrial Software Engineering, Quality Improvement Paradigm, Experimental Research Methods.

Nahid Shahmehri, Ph. D., Linköping 1991. Assistant professor (*universitetslektor*), computer science. On leave 1993-94 for Carlstedt Elektronik AB, Göteborg. Group leader, IISLAB.

Programming theory, programming languages, debugging tools, compiling technology, information management, business process modelling, CSCW.

Åke Sivertun, Ph. D., Umeå 1993. Research associate (forskarassistent) at LIBLAB.
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.



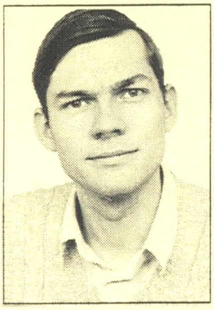
Toomas Timpka, MD., Stockholm 1983, Ph. D., Linköping 1989. Associate professor (*docent, universitetslektor*), computer and information science. Group leader, MDA. Acting Professor of Social Medicine and Associate Professor of Medical Informatics.

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.

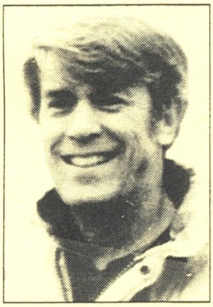
Guest Researchers Engaged in Graduate Study Program



Roland Bol, Ph. D., Amsterdam 1991. Guest researcher. Previous affiliation CWI, Amsterdam.

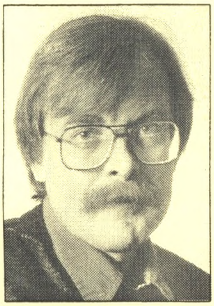
Logic programming, non-monotonic reasoning, deductive databases, process algebra.

Brant Cheikes, Ph.D., Univ. of Pennsylvania, 1991. Guest researcher.
Natural language processing and cooperative dialogue; architectures for response-planning systems, esp. intelligent help systems; simulator-based training systems. Applications of knowledge-based systems technology for operator training in the process industry.



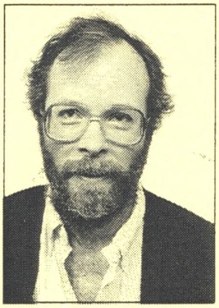
Robert L. Glass, M. Sc., University of Wisconsin 1954. Guest professor. President, Computing Trends. Previous affiliation Software Engineering Institute, Seattle University, The Boeing Company, USA. Honorary doctor at Linköping University 1995.

Software engineering: software quality, maintenance, design, technical communication, creativity. The importance of the application domain.



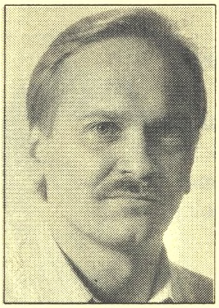
Witold Lukaszewicz, Ph. D., Warsaw University 1979. Guest professor. On leave from the Institute of Informatics, Warsaw University, Poland.

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.

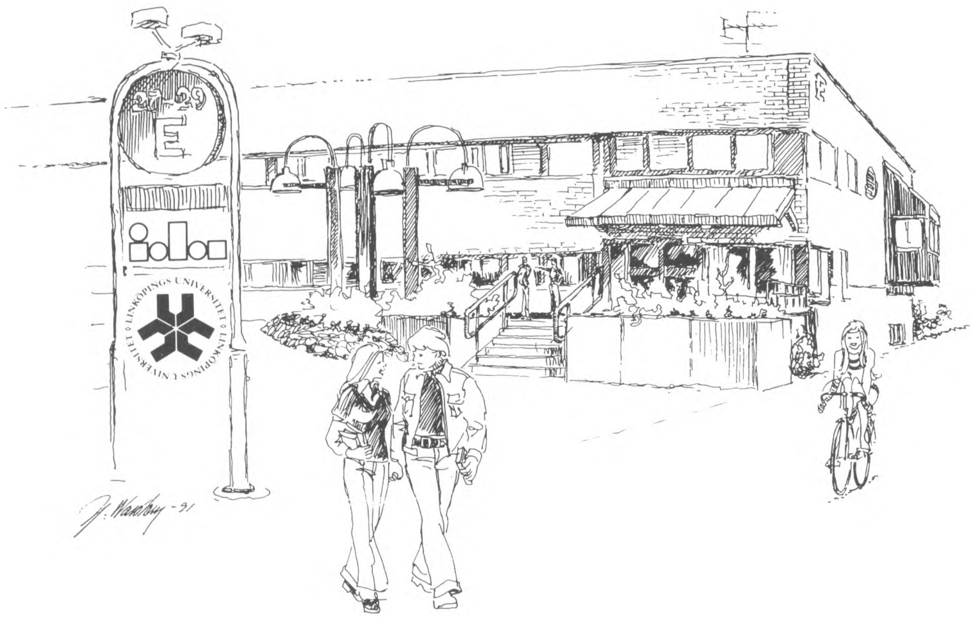


Jukka Paakki, Ph. D., University of Helsinki, 1991. Associate professor of computer science (University of Jyväskylä, Finland). Guest researcher.

Programming paradigms, language design and implementation, attribute grammars, logic programming.

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