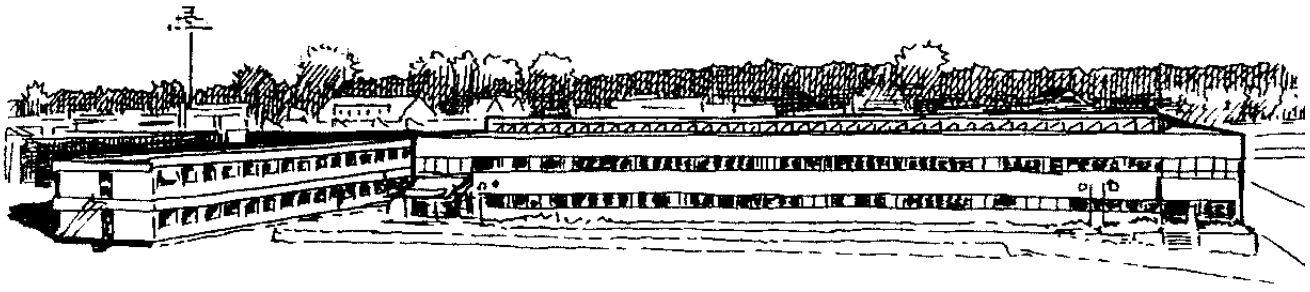
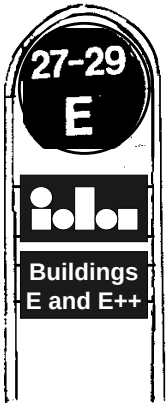
GRADUATE COURSES 92/93

DEPARTMENT OF COMPUTER AND INFORMATION SCIENCE

LINKÖPING UNIVERSITY





Institutionen för datavetenskap • Universitetet och Tekniska högskolan i Linköping • 581 83 Linköping
Forskn.sekr. Lillemor Wallgren • Telefon: 013/28 14 80 • Telefax: 013/14 22 31 • Elektronisk post: lew@ida.liu.se

Table of contents:

Linköping University
Department of Computer and Information Science
Lillemor Wallgren/*

June 25, 1992

The Graduate Program in Computer Science 1992 - 1993

This program describes the graduate courses planned for the academic year 1992 - 1993 within the areas dealt with by the Department of Computer and Information Science (Institutionen för datavetenskap, IDA): Computer Science, Computer Systems, Library and Information Science, Economic Information Systems, Computational Linguistics. Other courses and seminar series may be presented during the year, particularly intensive courses given by visiting researchers.

This program describes the following kinds of courses:

General Graduate Courses 1992--93
Recommended Masters courses
Laboratory-Oriented Activities, the Laboratories and the Faculty
Relevant Graduate Courses in Other Departments
Common Graduate Courses in the School of Engineering

The heading “Relevant Graduate Courses in Other Departments”, describes only courses for which information was available at the time when this program was printed. Other courses may be available. A comprehensive listing of all the graduate courses will be put out by the Planning Office in the fall.

Any changes to the course information will be spread by e-mail and notice-boards within IDA, through the IDA-COURIR, or by contacting directly those doctoral students enrolled in a course.

Inaugural Meeting of doctoral students: Tuesday, August 25, at 10.15 in the IDA Seminar room, 1st floor, E-building (see the inside back cover)

As far as possible, all doctoral students should take part in the following activities:
The main seminar series Some Tuesdays at 13.15.
Planned Seminars IDA/Tema in co-operation:
”Informationsteknologins användning och konsekvenser”. During the Fall: ”Kognitionsvetenskap”. During the Spring: ”Datavetenskapens historia i Sverige”.
Seminars are announced in IDA-COURIR, by e-mail, by the seminar file, and sometimes also by special announcement. Usually held in the IDA seminar room, 1st floor, E-building (see the inside back cover).

IDA-fika (Departmental coffee-break). Current information, short presentation of new arrivals in the department and visitors, descriptions of trips and conferences etc. **every Tuesday at 12.15 in the coffee area.**

Further information about the contents of this program can be obtained from Anders Törne, Director of Graduate Studies, phone 013 28 23 65 and Lillemor Wallgren, Administrator of Graduate Studies, phone 013 28 14 80 , or for a particular course, from the person responsible for the course within the Department of Computer and Information Science, Linköping University, S-581 83 Linköping.

1992/93

Cognitive Psychology

Recommended :
Graduate students Seminars 18h

The course last ran:
New course.

Goals:
To further develop some of the core topics in cognitive psychology, and to relate these to other areas in cognitive science.

Prerequisites:
Some basic knowledge in cognitive psychology is required, for example the undergraduate course TDDA 17 Psykologi, grundkurs (Psychology, introductory course) or something similar.

Organization:
A series of seminars where central articles in the field are presented and discussed. Active participation, including one article presentation, will give two points. An additional point will be given to those students that write a reviw paper on some course related topic.

Contents:
The focus will be on so-called higher cognitive processes, i.e. thinking, problem solving, language and communication.

Literature:
Articles.

Teachers:
Nils Dahlbäck

E*xaminer:*
Nils Dahlbäck

Schedule:
November - December, 1993.

Examination:
Seminar presentation, and paper

Credit:
2+1

1992/93

Data Structures and Graph Algorithms

Recommended for: *Lectures: 28h*
Computer Science students

The course last ran:
Spring 1990.

Goals:
The intent of the course is to reveal the interplay of data structures and new techniques for analysing algorithms in the design of efficient algorithms for four classical graph problems.

Prerequisites:
Basic course on algorithms and data structures, e.g. TDDB 57 Datastrukturer och algoritmer.

Organization:
Two lectures weekly, obligatory homework.

Contents:
1. Introduction: computational complexity, primitive data structures, etc..
2. Data structures for disjoint sets.
3. Heaps.
4. Search trees.
5. Linking and cutting trees.
6. Minimum spanning trees.
7. Shortest paths.
8. Network flows.
9. Matching.

Literature:
Robert E. Tarjan: Data Structures and Network Algorithms,
Regional Conference Series in Applied Mathematics, SIAM, 1983.

Teachers:
Per-Olof Fjällström.

Examiner:
Per-Olof Fjällström.

Schedule:
November - December, 1992.

Examination:
Obligatory homework.

Credit:
4 points.

1992/93

Fuzzy Logic and Control

Recommended for:
Computer Science and Systems students

Lectures: 24 h

The course last ran:
Has not been given previously.

Goals:
To familiarize students with the basic notions and ideas of fuzzy logic and set theory and their applications to intelligent control systems

Organization:
12 seminars

Contents:
- Fuzzy sets
- Fuzzy logic
- Possibilistic logic
- Applications to fuzzy control

Literature:
Lecture notes and articles

Teachers:
Dimiter Driankov

Examiner:
Dimiter Driankov

Schedule:
November - February, 1992/93

Examination:
Written term paper

Credit:
3 points

1991/92

Generation of Incremental Environments

Recommended:
For computer science and systems students

Lectures+seminars:
30h

Goals:

Knowledge of methods for specifying, generating and implementing incremental programming environments.

The course last ran:

This course has not been given before. However previous programming environment courses (1989/90 and 1986) contained approx. 10-20% of this material.

Prerequisites:

Some knowledge of compiler construction, attribute grammars and semantics.

Organization:

Eight introductory lectures followed by 7 seminars where course participants give presentations. Finally, a small implementation project of using one of the available systems to specify and generate a small language environment.

Contents:

General methods for implementing and generating incremental and language-oriented environments and editors. This includes *attribute grammar techniques* such as optimal attribute propagation techniques (Synthesizer Generator) and incremental attribute evaluation using *door attribute grammars* with reversible side-effects (the Mjölner system). Also language specification using *algebraic semantics*, and its use in specifying language syntax and semantics, and generating incremental environments including editing and execution support (the ASF+SDF system). Also, generation of language implementations from specifications in *natural semantics* (structured operational semantics), using the Centaur system. Finally, *unification-based incremental semantic analysis* that can handle incomplete program fragments, and can be used when generating language-oriented editors (the PSG system).

Literature:

Reps and Teitelbaum: The Synthesizer Generator. Bergstra, Heering, Klint: Algebraic Specification. Bahlke, Snelting: The PSG System - From Formal Language Definitions to Interactive Programming Environments, Görel Hedin: PhD thesis on incremental semantic analysis. Reports on Natural Semantics.

Teachers:

Peter Fritzson and invited lecturers

Examiner:

Peter Fritzson

Schedule:

ht-92

Examination:

Seminar presentation, writing a small specification in each formalism, and an optional programming project.

Credit:

4p (+ 1 -2 p)

1992/93

Information och verksamhet

Recommended for: *Lectures:* 24 h
I första hand för ekonomiska informationssystem, särskilt systemutveckling

The course last ran:
Ny kurs

Goals:
Öka den teoretiska förståelsen av informationssystem som socialt fenomen med utgångspunkt i teorier om information, kunskap, kommunikation, handling och verksamhet.

Prerequisites:
Önskvärt men inte nödvändigt är att man läst kursen Teorier och strategier för informationssystem innan. Denna kurs ges som SVL3-kurs i oktober med möjlighet för doktorander att följa den.

Organization:
Undervisningen ges i stark dialogkaraktär och bygger på att deltagarna läst litteratur i förväg. Avslutande seminarier.

Contents:
Teorier om information/kunskap och verksamhet/handling (fenomenologisk kunskapssociologi, talaktsteori, aktivitetsteori). Samspel information och verksamhet. Informationssystem och verksamhetsutformning. Kommunikation och kommunikativt handlande. Humaninfologisk teori om informationssystem.

Literature:
Berger & Luckmann: Kunskapssociologi; Artiklar/bokextrakt.

Teacher:
Göran Goldkuhl

Examiner:
Göran Goldkuhl

Schedule:
9/10 13-15, 2/11 13-17, 3/11 8-12, 23/11 13-17, 24/11 8-12; seminarietider tillkommer.

Examination:
Skriftlig rapport. Seminarium.

Credit:
3 + 3p

Introduction to research methodology in computer science

Recommended for:
New graduate students

Lectures:16 h

Last offering:
This course is a slightly modified version of a subset of the course given in 90/91.

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 sience. 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.

Teacher:
Sture Hägglund, et al.

Examiner:
Sture Hägglund

Schedule:
September-October, 1991.

Examination:
Written examination and seminar activity.

Credit:
2 points

1992/93

Introduction to research methodology in computer science

Recommended for:
New graduate students

Lectures:16 h

Last offering:
1992/93.

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 sience. 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.

Teacher:
Sture Hägglund, et al.

Examiner:
Sture Hägglund

Schedule:
September-October, 1992.

Examination:
Written examination and seminar activity.

Credit:
2 points

Introduction to Systems Theory

Recommended for:
Computer Sciences and Systems students

Lectures: 24 h

The course last ran:

A new course

Goals:

To give an introductory **formal** presentation of 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) interested in basic control theory concepts, and computer systems students (D-line) 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.

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 (response 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;

Literature:

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:

Autumn 92 (September - October, twice a week)

Examination:

Written assignments throughout the course. Term paper.

Credit:

3 points

1992/93

Natural Language Processing

Recommended for:
Graduate students

Lectures: 20 h

The course last ran:
Spring 1989

Goals:
To give an overview of important concepts, methods and issues in natural language processing, centered around understanding.

Prerequisites:
Basic knowledge of logic and AI.

Organization:
There will be 10 lectures given by the teacher and two seminars with presentations by participants.

Contents:
Knowledge representation for natural language: lexicon, grammar, discourse and world knowledge. Understanding: parsing, semantic interpretation, reference resolution and disambiguation in context. Generation. Applications of NLP.

Literature:
James Allen: *Natural Language Understanding.* Menlo Park, Benjamin/Cummings, 1987 and articles. (Allen's book may be replaced, if something good and more up-to-date appears during this year.)

Teacher:
Lars Ahrenberg.

Examiner:
Lars Ahrenberg.

Schedule:
October - November, 1992

Examination:
An essay and oral presentation of an application area, or problem area of NLP.

Credit:
3 points

1992/93

Petri Nets and Formal Description of Parallel Systems

Recommended for:
Computer Science and System students

Lectures: 24 h

The course last ran:
This is a modified version of the course ”Petri Nets and their Applications” given in 89/90.

Goals:
To give a basic knowledge of Petri nets and an overview of different computation models of parallel systems. The emphasis will be on formal notations to express concurrency and various properties of parallel systems that can be studied with these notations.

Prerequisites:
Basic courses in Discrete Mathematics and some basic knowledge of Formal Languages and Automata Theory.

Organization:
The course will consist of lectures, seminars given by the participants, and obligatory homework.

Contents:
- Overview of parallel computation models.
- Introduction and basic definitions of the Petri net theory.
- Modeling systems with Petri nets.
- Analysis, complexity and decidability issues.
- Extended and restricted Petri net models.
- Other models of parallel computation: CCS and CSP.
- Application of formal description models.

Literature:
James L. Peterson, ”Petri Net Theory and the Modeling of Systems”, Prentice-Hall, 1981. Copies of articles describing other parallel models will be distributed during the course.

Teachers:
Zebo Peng and Krzysztof Kuchcinski

Examiner:
Zebo Peng

Schedule:
November - December, 1992.

Examination:
Obligatory homework. Additional points will be given to those who will carry out in-depth studies and write a term paper.

Credit:
4 + 2 points

Principles of Modern Database Systems

Recommended for: *Lectures* 40h
Graduate students

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

Prerequisites:
Undergraduate courses in computer science. Basic database course

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

Contents:
- Overview Traditional Database Systems
- Advanced Database Applications
- Data Models (e.g. relational, OO, functional)
- Object Data Management Concepts
- OO Query Languages
- Concurrency and Recovery
- Versioning
- Database Performance and Benchmarks
- Heterogeneous Databases
- Active Databases
- Rules in Databases
- Database Procedures
- Deductive Databases
- Temporal Databases
- Main-memory Databases

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

Teachers:
Tore Risch

Schedule:
HT 1992

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

Credit:
5 + 2 points

1992/93

Redovisningsteori

Recommended for:
Forskare med ekonomisk bakgrund.

Lectures: ca 40 h

The course last ran:
Ny kurs.

Goals:
Externredovisningsområdet är idag mycket splittrat såväl vad avser praxis som rekommendationer och teoretiska ansatser. Mot denna bakgrund syftar kursen i redovisningsteori till att presentera grundläggande utgångspunkter inom externredovisningen för att möjliggöra en förståelse för och kritisk analys av dagens externredovisning. En sådan förståelse är nödvändig för värdering och utveckling av praxis inom området.

Organization:
Föreläsningar och seminarier.

Literature:
Meddelas senare.

Teachers:
Jörgen Dahlgren

Examiner:
Birger Rapp

Schedule:
v36, v41, v42, v49, v50 enligt schema.

Examination:
Skriftlig tentamen och inlämningsuppgifter.

Credit:
5 poäng

1992/93

Vetenskaplig metodik

Recommended for:
Alla forskarstuderande

Lectures:

The course last ran:
Ny kurs

Goals:
Att ge tillräcklig vetenskaplig grund för avhandlingsarbetet.

Organization:
Föreläsningar och laborationer

Contents:
Kvantitativ vetenskaplig metodik med bl a systemanalys, statistik och simuleringsteknik.

Literature:
Se särskild lista.

Teachers:
Claes Wahlbin, Leif Melin, Sten Wandel, Birger Rapp, Eva Leander, Bo Sellstedt, Staffan Brege, Evert Vedung m fl.

Examiner:
Birger Rapp

Schedule:
Avslutas v34, v35 och v36 enligt särskilt schema.

Examination:
Tentamen, inlämningsuppgifter.

Credit:
10 poäng totalt (går att få delpoäng).

1992/93

Advanced Logic

Recommended for:
Graduate students.

Lectures: 24 h

Goals:
The purpose of the course is to develop an intuitive grasp of the main results of modern logic and to explain their relationship to computer science, particularly artificial intelligence and logic programming.

Prerequisites:
TDDA 15 Logic I, (TDDA 16 AI-Knowledge Representation suggested) or Logic for IDA-ites.

Organization:
8 three-hour seminars, once a week over 8 weeks. Each seminar will consist of a two-hour presentation followed by an hour's discussion. Independent reading, approx. 4 hours each week.

Contents:
- Löwenheim - Skolem Theorem
- Tarski semantics
- Skolem-Herbrand-Gödel Theorem
- Sequent calculus
- Gentzen's Theorem
- Church's Theorem
- Gödel's Incompleteness Theorem

Literature:
E. Mendelson, Introduction to Mathematical Logic, Wadsworth 1987
J Gallier, Logic for Computer Science, Wiley (paperback)

Teacher:
Douglas Busch

Examiner:
Douglas Busch

Schedule:
April - May 1993.

Examination:
A written report on a chosen topic.

Credit:
Up to 4 points.

Argumentation Theory: Part I Philosophical Foundations

Recommended: *Lectures: 24 h, Seminars 6h*
Graduate students

The course last ran:
New course.

Goals:
To give a theoretical and philosophical background for the study of reasoning and argumentation. The course provides a basis for further studies in modal, non-standard, non-montonic and non-formal logics and the analysis and generation of argumentation in natural language which is the topic of the course Argumentation Theory: Part II Artificial Intelligence and Natural Language Processing Applications.

Prerequisites:
Logic for IDAites is recommended but not necessarily required.

Organization:
The course will begin with classical analyses of reasoning and argumentation in natural language and move toward modern attempts to account for the ’logical’ structure of everyday (non-mathematical and logical) reasoning and argumentation.

Contents:
Pro et Contra, Pro ut Contra reasoning, Dialectics, Dialogic logic, Plausible inference, Topical reasoning, Fallacies, Presupposition, Implicature, Mills Methods, Modal reasoning, Situational and Field properties of reasoning and argumentation.

Literature:
Articles

Teachers:
Richard Hirsch

Examiner:
Richard Hirsch

Schedule:
January - March 93

Examination:
Written or oral exam and seminar presentation

Credit:
3 + 1 points.

1992/93

Automated Debugging

Recommended for:
Graduate students

Lectures:
20 h

The course last ran:
This course has not been given before.

Goals:
The goal of the course is to give an overview of various approaches to automated debugging.

Prerequisites:
None.

Organization:
Three introductory lectures followed by seven seminars, where the course participants present papers.
A small one-week implementation project gives an extra point.

Contents:
- How far are we from the dream of automatic debugging.
- Requirements for the implementation language.
- Knowledge requirements.
- Various approaches to generate automated debugging systems.
- Existing automated debugging systems.

Literature:
Scientific Papers

Teachers:
Nahid Shahmehri

Examiner:
Nahid Shahmehri

Schedule:
January-march, 93.

Examination:
Seminar presentation and written report.
Implementation project gives extra point.

Credit:
3p (+1p)

1992/93

Computability and Complexity Theory

Recommended for: *Lectures: 28h*
Computer Science students.

The course last ran:
The content of the course is similar to that of TDDA 74 Beräkningsbarhet och Komplexitet, 1988.

Goals:
The course addresses the limitations of what can be achieved by computers. We show that there are, important and mathematically well-defined, problems which are either unsolvable or the solution of which would require formidable amounts of computer resources.
However, these negative results can also be used to our advantage. The last part of the course shows that the existence of intractable problems can help us design good cryptographic schemes.

Prerequisites:
Course on Discrete Mathematics.

Organization:
Two lectures weekly, obligatory homework.

Contents:
1. Turing Machines and the Church/Turing Thesis.
2. Basic undecidability results.
3. Degrees of undecidability.
4. Post correspondence problem and Hilbert’s tenth problem.
5. Axiomatic complexity theory.
6. Complexity classes and intractability.
7. Applications of complexity theory in cryptography.
8. Interactive proofs and zero-knowledge protocols.

Literature:
Arto Salomaa: Computation and Automata,
Cambridge University Press, 1985.

Teachers:
Per-Olof Fjällström

Examiner:
Per-Olof Fjällström

Schedule:

April - May, 1993.

Examination:
Obligatory homework.

Credit:
4 points.

1992/93

Computers and Computer Science: A Historical Approach

Recommended for:
Graduate students

Lectures: 20 h

The course last ran:
This is a new course.

Goals:
This course will attempt to tie the history of computing and computer science to studies of history and society. In particular, we will use material from the history of science and technology to examine some of the evolutions, revolutions, and transitions (e.g., from analog to digital) that have characterized the development of computer technology and computer science as an academic discipline.

Prerequisites:
None.

Organization:
Lectures and seminars.

Contents:
To be covered are transfer (of problem and practice) between computer science, science, and engineering, and between academia, government and industry. Material from Sweden and the US will be used.

Literature:
To be assigned.

Teachers:
J. Nyce

Examiner:
J. Nyce

Schedule:
April-June, 1993.

Examination:
Seminar participation and paper.

Credit:
3 points.

Introduction to Neural Nets

Recommended for: *Lectures:* *24 h*
Computer science and system students

The course last ran:
It is a new course.

Goals:
The course will give a basic knowledge about theory and practice of neural networks. The basic neural networks structures will be discussed together with different architectural solutions for their implementations.

Prerequisites:
Basic knowledge in mathematical analysis.

Organization:
The course will consist of lectures. The seminars on interesting topics can be given by participants.

Contents:
Fundamentals of artificial neural networks.
Perceptrons.
Backpropagation.
Counterpropagation networks
Statistical methods
Hopfield networks
Bidirectional associative memories
Adaptive resonance theory
Optical neural networks
The cognitron and neocognitron
Massively parallel computers in neural networks implementation

Literature:
Philip D. Wasserman, “Neural Networks, Theory and Practice”, Van Nostrand Reinhold, New York.

Teachers:
Krzysztof Kuchcinski

Examiner:
Krzysztof Kuchcinski

Schedule:
March- May 1993. Thursday 10:15- 12:00

Examination:
Obligatory homework.

Credit:
3 points

Intelligent Information Systems

Recommended for: *Lectures:* 24 h
Graduate students - suitable for students from all IDA programs.

The course last ran
New course

Goals:
To introduce the area of Intelligent Information Systems, place it in a context and give an understanding of the kind of research within the area. (A course which contributes to breadth rather than depth of the graduate education).

Prerequisites:
None necessary. Some basic knowledge of computer science, in particular A. I. is useful.

Organization:
The course will consist of some introductory lectures giving an overview and context, followed by a series of seminars where course participants are active in presenting current research projects in I.I.S. There will be a base collection of papers to be read by all participants. Each participant will then choose one system or area for deeper investigation and presentation.
Contents:
History and scope of I.I.S.
D.B. - A.I. interface
Some I.I.S. visions
Some I.I.S. research projects
Some aspects/issues of I.I.S. research

Literature:
Collection of research papers

Teachers:
Lin Padgham, Martin Sjölin (assistant)

Examiner:
Lin Padgham

Schedule:
Spring 1993

Examination:
Short summary of required reading to be handed in at each lecture. A 5-10 page report of some I.I.S. system or issue

Credit:
3 points

1992/93

Knowledge Acquisition

Recommended for:
Graduate students

Lectures: *24 h*

The course last ran:
New course.

Goals:
To structure techniques applicable in acquiring knowledge for knowledge-based systems.

Prerequisites:
Basic knowledge about system development and artificial intelligence.

Organization:
Lectures (11x2h) and a role-game exercise (home-work) plus a seminar (1x2h).

Contents:
- Knowledge-based systems development
- Working with the expert
- Interviewing techniques
- Automated tools for knowledge acquisition
- Validation of knowledge-based software

Literature:
McGraw, K. L. and Harbison-Briggs, K., *Knowledge Acquisition: Principles and Guidelines,* Prentice Hall, 1989.
Selected papers.

Teachers:
Kristian Sandahl and guests

Examiner:
Kristian Sandahl

Schedule:
Spring 1993.

Examination:
Active participation in role-game with a group.
Attendance at the final seminar where experiences from the role-game exercise are presented.
Written examination showing an overall familiarization with the area.
Term paper showing a deep understanding of a selected sub-topic.

Credit:
3 points for the attendance, role-game participation and the written exam
3 points for a high quality term paper

1992/93

Kvalitativ undersökningsmetodik

Recommended for: *Lectures:* 24 h
I första hand för ekonomiska informationssystem, särskilt systemutveckling

The course last ran:
Ny kurs

Goals:
Öka förståelse avseende behovet av, tillvägagångssätt vid och problem med kvalitativa undersökningar. Öka förmågan att genomföra kvalitativa undersökningar.

Prerequisites:
Grundläggande kurs i forskningsmetodik; tex Kunskapsutveckling om informationssystem.

Organization:
Undervisningen ges i stark dialogkaraktär och bygger på att deltagarna läst litteratur i förväg. Avslutande seminarier.

Contents:
Olika ansatser, strategier och metoder för kvalitativa undersökningar: Deltagande observation, kvalitativa intervjuer, aktionsforskning, participativ forskning, fallstudie-metodik, hermeneutiska principer för tolkning av källor, empiribaserad teori-utveckling (grounded theory). Närhet, access och engagemang vs distans och reflektion. Förförståelse och öppenhet.

Literature:
Gummesson: Forskare och konsult - om aktionsforskning och fallstudier i företags-ekonomin, Studentlitteratur
Starrin m fl: Från upptäckt till presentation, Studentlitteratur
Repstad: Närhet och distans, Studentlitteratur
Ödman: Tolkning, förståelse, vetande.Hermeneutik i teori och praktik, AWE/Gebers

Teachers:
Göran Goldkuhl

Examiner:
Göran Goldkuhl

Schedule:
Börjar preliminärt i januari 93

Examination:
Skriftlig rapport. Frivillig del 2 innebär genomförande av mindre kvalitativ undersökning.

Credit:
3 + 3p

Object-Oriented Programming

Recommended for:
Graduate students

Lectures: 24 h

The course last ran:
Partial overlap with ”Object-Oriented Programming and Design Methodology”, 1990/91. The main difference is that this time especially language issues and programming techniques will be emphasized.,

Goals:
To present the basic and advanced concepts of object-oriented programming and to develop programming skills in the language Eiffel.

Prerequisites:
Familiarity with the material of TDDA 33: ”Objektorienterad programmering” may be helpful, but it is not obligatory.

Organization:
Lectures, programming project.

Contents:
The following concepts and primitives of object-oriented programming and Eiffel will be introduced:

- Abstract data types
- Classes and objects
- (Multiple) inheritance
- Deferred classes and abstract operations
- Prototype languages and delegation
- Generics
- Polymorphism and dynamic bindning
- Assertions

Literature:
Bertrand Meyer: Object Oriented Software Construction, Prentice-Hall, 1988.
Articles.

Teachers:
Jukka Paakki

Examiner:
Jukka Paaki

Schedule:
Spring 1993.

Examination:
Written examination and a programming project.9393

Credit:
3 points

1992/93

Structured Operational Semantics

Recommended for:
Graduate students.

Lectures: 18 h

The course last ran:
New course.

Goals:
To present a simple method for specifying the semantics of programming languages based on Plotkins structural operational semantics. To show applications to functional and logic programming and imperative languages with various parameter passing schemes. Show how to use the semantic definition for static analysis (abstract interpretation), transformation and verifying correctnes of programs.

Prerequisites:
Discrete mathematics.

Organization:
Ten lectures.

Contents:
Semantics, Models of execution, Transition systems, Verification, Analysis, Transformation.

Literature:
Gordon Plotkin: A Structural Approach to Operational Semantics. DAIMI fn-19, Aarhus University (1981) and articles.

Teachers:
Ulf Nilsson

Examiner:
Ulf Nilsson

Schedule:
Spring 1993

Examination:
Home assignments

Credit:
3 points.

1992/93

The Role of Maintenance in Software Engineering

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

The course last ran:
New course.

Goals:
Develop an understanding of, and practice in, software's most cost- and activity-consuming phase.

Prerequisites:
Software Engineering Design and Methodology.

Organization:
Lectures and discussion, group projects, individual homework.

Contents:
One of the underlying requirements for almost all software is that it must be maintainable.
This course answers these questions:
What is maintainable software? Why do we care? How do we build it? How do we do maintenance? How can we do it better in the future? What is the relationship between quality and maintenance? The answers will come through interactive discussion, through lectures, and through workshop experiences.

Literature:
A collection of readings.

Teachers:
Robert L. Glass.

Examiner:
Robert L. Glass.

Schedule:
May - June, 1993.

Examination:
Group project, individual exercises, open-book exam.

Credit:
3 points.

1992/93

Svensk och internationell externredovisning

Recommended for:
Forskare med ekonomisk bakgrund

Lectures: ca 40 h

The course last ran:
Ny kurs

Goals:
Att utifrån svensk externredovisning ge insikter och kunskap om internationell externredovisning och förväntade framtida utvecklingar inom framförallt EG.

Organization:
Föreläsningar och seminarier.

Literature:
Meddelas senare.

Teachers:
Rolf Rundfeldt.

Examiner:
Rolf Rundfeldt.

Schedule:
v5, veckor i maj och 2 veckor i juni enligt schemat.

Examination:
Skriftlig tentamen och inlämningsuppgifter.

Credit:
5 poäng.

Recommended Master Courses

Under Recommended Master Courses:
Teorier och strategier för informationssystem, 3p, SVL3 Datorstödd systemutveckling, 4p, SVL3

Laboratory-Oriented Activities

The Laboratories and the Faculty

LABORATORY-ORIENTED COURSES

Like the general gruduate courses these are open for all doctoral students, but they are organized so as to have a direct link to activites in each laboratory.

Laboratory for Complexity of Algorithms (ACTLAB)

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

Development of Information Systems and Work Activities (VITS)

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

Two courses will be given namely ”Kunskapsutveckling om informationssystem” och ”Förändringsanalys”. See separate course descriptions.

Application Systems Laboratory (ASLAB)

Software Engineering of Object-Oriented Systems. Interest group/study circle.

Simulation technique for decision making and training. Some Thursdays between 13 -15**.**

Fixed time for lab meetings: **Thursdays between 13-15.**

Computer Aided Design Laboratory (CADLAB)

Fixed time for lab meetings: **Thursdays between 13-15.**

Laboratory for Computer Assistance in Engineering (CAELAB)

Fixed time for lab meetings: **Wednesdays between 15-17.**

Economic Information Systems (EIS)

No fixed time for lab meetings.

Library and Information Science Laboratory (LIBLAB)

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

Laboratory for Intelligent Information Systems (IISLAB)

Fixed time for lab meetings: **Wednesdays between 10 -12.**

Laboratory for Logic Programming (LOGPRO)

Fixed time for lab meetings: **Wednesdays between 15-17.**

People, Computers and Work (PCW)

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

Natural Language Processing Laboratory (NLPLAB)

Fixed time for lab meeting**s: Wednesdays between 15-17.**

Programming Environments Laboratory (PELAB)

Fixed time for lab meeting**s: Wednesdays between 15 - 17.**

Laboratory for Representation of Knowledge in Logic (RKLLAB)

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

Research laboratories and groups

The department is led by a Department Board (institutionsstyrelse). The Department Chairman (“prefekt”) is Anders Haraldsson, as from July 1, 1990. Vice Chairman is Erik Sandewall.

The two main areas of activity are reflected in two subordinate committees,:

- The Undergraduate Teaching Committee (IDUN), whose Chairman is Olle Willen.

.

- The Research Committee (FND), whose Chairman is Sture Hägglund.

The Research Committee members are appointed by the Department Board, and the committee approximately equals the set of laboratory leaders. It is responsible for all aspects of the department’s graduate education programs and research.

The responsibility for managing the graduate education program rests with the Research Secretariat, made up of Sture Hägglund, Chariman of the Research Committee, Anders Törne, Director of Graduate Studies (forskarstudierektor) and Lillemor Wallgren, Administrator of Graduate Studies.

The graduate research at IDA is organized into the following laboratories/groups:

ACTLAB – Laboratory for Complexity of Algorithms

Per-Olof Fjällström

which 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 together with the Mechanical Engineering department in the context of CENIIT, is efficient algorithms for detection of contacts.

ASLAB – Application Systems Laboratory

Sture Hägglund

which is oriented towards the study of knowledge-based approaches to software development, and of certain aspects of human-computer interaction. Major areas of interest in the lab are methods and tools for user interface design (Jonas Löwgren) and expert critiquing systems. Joint projects involve cooperation with industry in the knowledge transfer programme, and with several other research groups. A special subproject studies *Industrial Software Technology*, including object-oriented software engineering and knowledge-based systems (Kristian Sandahl).

CADLAB – Laboratory for Computer-Aided Design of Digital Systems

Krzysztof Kuchcinski

which 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.

CAELAB – Laboratory for Computer Assistance in Engineering

Anders Törne, Tore Risch

which is concerned with information technology to be used in industrial processes and products, in particular computer support for automation and real-time systems. Engineering database technology is an area of special interest (Tore Risch). This lab has its major funding from CENIIT and NUTEK. It is heavily engaged in cooperative efforts with for instance the Mechanical Engineering Department.

EIS – Economic Information Systems

Birger Rapp

which 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 resource management, business control and decision support systems. Research and graduate education is also coordinated with corresponding activities in the administrative data processing group.

IISLAB – Laboratory for Intelligent Information Systems

Lin Padgham

which studies theory and methods for distributed information management, including object-orientation and inheritance strategies. In a major implementation project, an object-oriented database, with a distributed architecture, support for history maintenance and parallel editing, has been developed.

LIBLAB – Laboratory for Library and Information Science

Roland Hjerppe

which 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. Recently new technologies, like hypermedia, and new methods of scientific communication and knowledge organization have become of primary interest to LIBLAB.

LOGPRO – Laboratory for Logic Programming

Jan Maluszynski

which has its research concentrated on foundations of logic programming, relations to other programming paradigms and methodology. The major current projects are concerned with “Logic Programming and External Procedures” and with “Systematic Design of Abstract Machines through Partial Evaluation”.

MDA – People, Computers and Work

Toomas Timpka

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 in 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

which conducts research related to the development and use of natural language interfaces to computer software. The interests of the lab covers most aspects of natural language processing and computational linguistics, with theoretical research interests primarily in parsing and interpretation, in knowledge representation for NL understanding and in the characteristics of man-machine NL interaction.

PELAB – Programming Environments Laboratory

Peter Fritzson

which works with design of tools for software development, specific functions in such tools and theoretical aspects of programs under construction. Projects are focused on two areas, namely on development of an architecture for a programming environment supporting design, implementation and maintenance of distributed systems, and on development and evaluation of tools for meta-level language support in a multi-level software architecture. A major undertaking is participation in the ESPRIT GIPE II project.

RKLLAB – Laboratory for Representation of Knowledge in Logic

Erik Sandewall

which covers issues and techniques such as non-monotonic logic, probabilistic logic, modal logic and truth maintenance algorithms and systems. Research in the lab combines fundamental theory development with practical work in applied projects. An important theme is the study of *plan-guided systems*, with applications for instance in mission planning for autonomous vehicles, with external cooperation in for instance the European Prometheus (future road traffic) project.

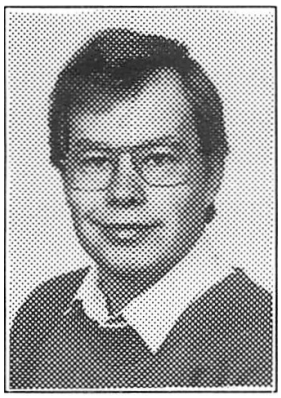
VITS – Development of Information Systems and Work Activities

Göran Goldkuhl

which 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 information systems evaluation. Strategies for information systems architecture. Relations between methods and CASE tools (CASE shells).

Appendix B
Graduate Study Program

Faculty presently engaged in graduate study program.

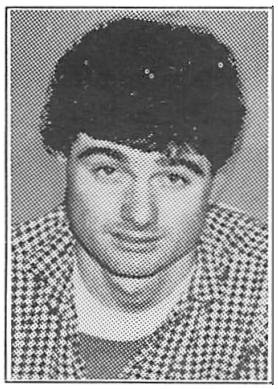


Lars Ahrenberg, PhD, Uppsala 1987. Associate professor (*docent, högskolelektor*) of computational linguistics. Group leader, NLPLAB. Previous affiliations Uppsala and Göteborg.

Syntax, semantics and pragmatics of natural language; natural language understanding; natural language interfaces; text generation.

Douglas Busch, PhD, Rockefeller 1973. Associate professor (*högskolelektor*) of logic and theoretical computer science. Previous affiliation Mcquarie Univ., Sydney, Australia. Director of the graduate study programme.

Application of theories from formal logic to problems in theoretical computer science and artificial intelligence; semantics of logic programs; philosophical questions in artificial intelligence.



Keith Downing, PhD, Univ. of Oregon 1990. Guest researcher.

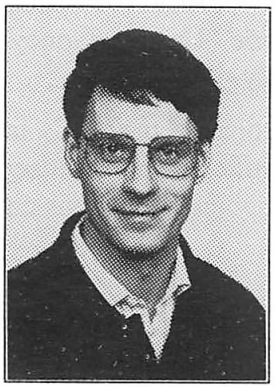
Model-based diagnosis, qualitative physics, reason maintenance, artificial intelligence in medicine. Applications of model-based simulation and diagnosis to physiological domains

Dimiter Driankov, PhD, Linköping 1989. Assistant professor *(högskolelektor)*, logic and AI.

Reasoning under uncertainty, many-valued logics, knowledge-based plan-recognition, decision support systems.

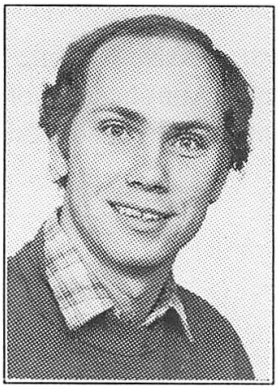
Johan Fagerström, PhD, Linköping 1988. Assistant professor (*högskolelektor*), computer science. Director of undergraduate studies in System Architecture.

Distributed systems, parallel systems, operating systems.



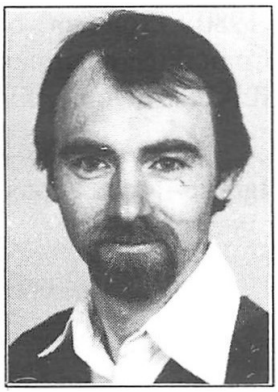
Per-Olof Fjällström, PhD, Stockholm 1985. Associate professor *(högskolelektor)*, theoretical computer science. Previous affiliation KTH and IBM. Group leader, ACT-LAB.

Computational geometry, analysis of algorithms, data structures.



Peter Fritzson, PhD, Linköping 1984. Associate professor (*högskolelektor*), computer science. Group leader, PELAB. Previous affiliations Sun MicroSystems USA 1985-86, Linköping Univ. Hospital 1976-79.

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



Göran Goldkuhl, PhD, Stockholm 1980. Associate professor (*docent, högskolelektor*), administrative data processing. Group leader in systems development research. Previous affiliation Göteborg.

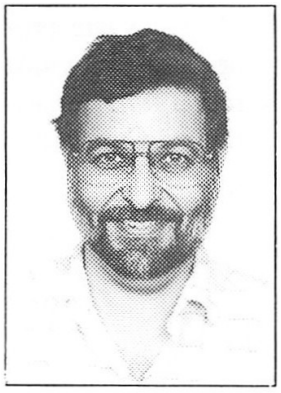
Information requirement analysis, behavioral aspects of information systems, research methodologies, information systems and quality of working life.

Anders Haraldsson, PhD, Linköping 1977. Associate professor (*högskolelektor*), computer science. Department chairman (from autumn 1990) and Undergraduate Committée chairman (until autumn 1990). Previous affiliation Uppsala.

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

Richard Hirsch, PhD, Gothenburg 1989. Lecturer (*högskoleadjunkt*) 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; philosophy of language.



Roland Hjerppe. Researcher. 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, PhD, Linköping 1980. Professor of knowledge-based systems. Research Committee chairman (from autumn 1990). Group leader, ASLAB. Previous affiliation Uppsala.

Expert systems and artificial intelligence applications, database technology, human-computer interaction.

Feliks Kluzniak, PhD, Warsaw University 1987. Guest researcher. On leave from the Institute of Informatics, Warsaw University.

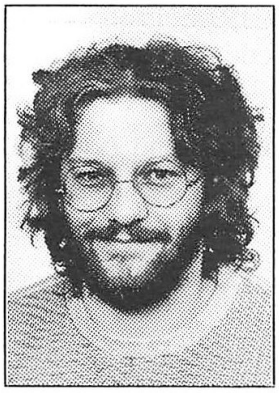
Logic programming, compilers, programming methodology.

Krzysztof Kuchcinski, PhD, Gdansk 1984. Acting professor of computer systems. Group leader, CADLAB. Previous affiliation Technical Univ. of Gdansk, Poland.

Computer architecture, computer-aided design of digital systems, VLSI, test generation methods.

Bengt Lennartsson, PhD, Göteborg 1974. Associate professor (*högskolelektor*), software engineering. Previous affiliation Luleå. Group leader, PELAB until 1988. Department chairman 1983-1990. Undergraduate Committee chairman (autumn 1990).

Real-time systems, industrial software technology, large scale software development.



Jacek Malec, PhD, Wroclaw 1987. Assistant professor. Previous affiliation Technical Univ. of Wroclaw, Poland.

Artificial Intelligence: knowledge representation, planning, reactive systems, autonomous systems architecture, dynamic scene description.

Jan Maluszynski, PhD, Warszawa 1973. Professor of programming theory. Several previous affiliations. Group leader, LOGPRO.

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

Lin Padgham, PhD Linköping 1989. Assistant professor *(högskolelektor)*, computer science. Previous affiliation Univ. of Oregon, USA, and Tektronix. Group leader, IISLAB.

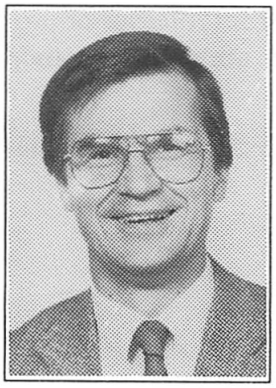
Inheritance, default reasoning, taxonomical reasoning, object-oriented systems.

Zebo Peng, PhD, Linköping 1987. Assistant professor (*högskolelektor*), computer architecture.

Automated synthesis of digital systems, formal description of hardware, VLSI, computer-aided design, computer architecture.

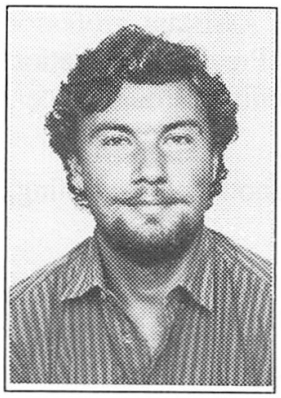
Birger Rapp, Econ. Dr., Stockholm 1974, Professor of Economic Information Systems. Several previous affiliations.

Accounting systems, economic control, IT and organisation, production, economics.



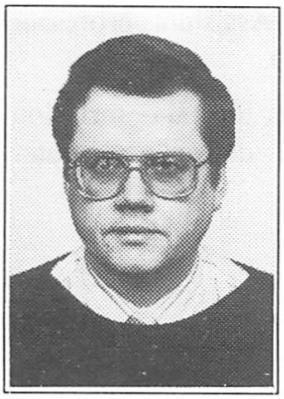
Erik Sandewall, PhD, Uppsala 1969. Professor of computer science. Research Committee chairman (1983-1990). Group leader, RKLLAB. Several previous affiliations.

Representation of knowledge with logic, autonomous agents, knowledge-based planning.



Toomas Timpka, MD., Stockholm 1983, PhD, Linköping 1989. Assistant professor (*högskolelektor*), computer and information science. Group leader, MDA.

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



Anders Törne PhD, Uppsala 1980. Associate professor (*högskolelektor)*, computer support in automation. Group leader, CAELAB. Previously at ABB Corporate Research, Västerås.

Computer support for generation, transformation, and use of information in manufacturing processes. Architectures for processing control and supervision. Robot programming.

Relevant Graduate Courses in Other Departments

Common Graduate Courses in the School of Engineering

Vetenskapsteori

Presentationsteknik (in English)

Teknikhistoria