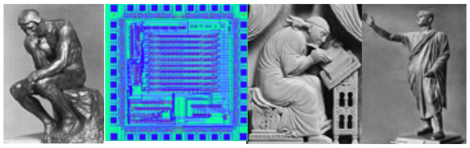
CT3620
VETENSKAPSMETODIK FÖR TEKNIKOMRÅDET

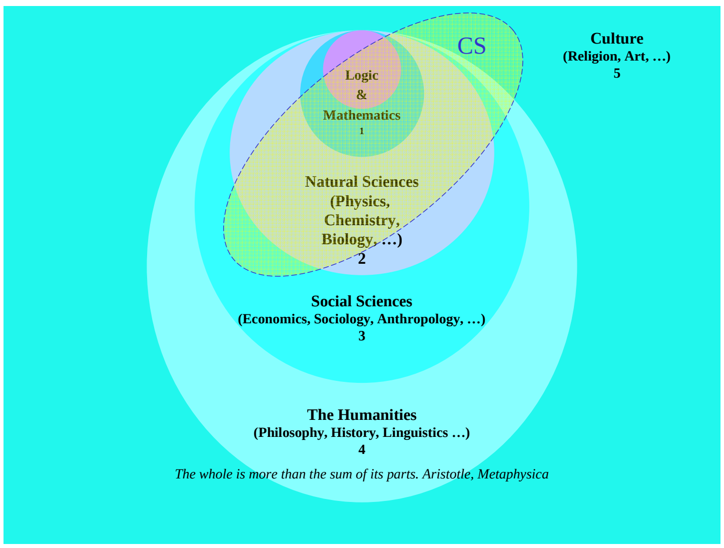
GRUNDLÄGGANDE VETENSKAPSTEORI



Gordana Dodig-Crnkovic
Department of Computer Science and Engineering
Mälardalen University

HISTORY OF COMPUTER SCIENCE

- LEIBNIZ: LOGICAL CALCULUS
- BOOLE: LOGIC AS ALGEBRA
- FREGE: MATEMATICS AS LOGIC
- CANTOR: INFINITY
- HILBERT: PROGRAM FOR MATHEMATICS
- GÖDEL: END OF HILBERTS PROGRAM
- TURING: UNIVERSAL AUTOMATON
- VON NEUMAN: COMPUTER

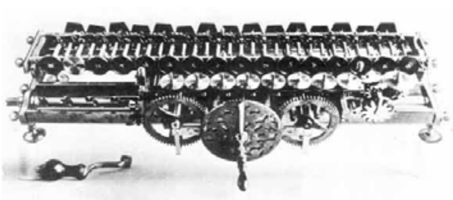


LEIBNIZ: LOGICAL CALCULUS

Gottfried Wilhelm von Leibniz

Born: 1 July 1646 in Leipzig, Saxony (now Germany)
Died: 14 Nov 1716 in Hannover, Hanover (now Germany)

LEIBNIZ´S CALCULATING MACHINE



"For it is unworthy of excellent men to lose hours like slaves in the labor of calculation which could safely be relegated to anyone else if the machine were used."

LEIBNIZ´S LOGICAL CALCULUS

DEFINITION 3. A is in L, or L contains A, is the same as to say that L can be made to coincide with a plurality of terms taken together of which A is one. B ⊕ N = L signifies that B is in L and that B and N together compose or constitute L. The same thing holds for larger number of terms.

AXIOM 1. B ⊕ N = N ⊕ B.

POSTULATE. Any plurality of terms, as A and B, can be added to compose A ⊕ B.

AXIOM 2. A ⊕ A = A.

PROPOSITION 5. If A is in B and A = C, then C is in B.

PROPOSITION 6. If C is in B and A = B, then C is in A.

PROPOSITION 7. A is in A.

(For A is in A ⊕ A (by Definition 3). Therefore (by Proposition 6) A is in A.)

….

PROPOSITION 20. If A is in M and B is in N, then A ⊕ B is in M ⊕ N.

BOOLE: LOGIC AS ALGEBRA

George Boole

Born: 2 Nov 1815 in Lincoln, Lincolnshire, England
Died: 8 Dec 1864 in Ballintemple, County Cork, Ireland

George Boole is famous because he showed that rules used in the algebra of numbers could also be applied to logic.

- This logic algebra, called Boolean algebra, has many properties which are similar to "regular" algebra.
- These rules can help us to reduce an expression to an equivalent expression that has fewer operators.

Properties of Boolean Operations

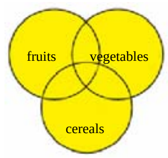
$A \cdot 0$	=	0
$A + 0$	=	A
$A \cdot 1$	=	A
$A + 1$	=	1
$A \cdot A$	=	A
$A + A$	=	A
$A \cdot \overline{A}$	=	0
$A + \overline{A}$	=	1

$\overline{\overline{A}}$	=	A
$A \cdot B$	=	$B \cdot A$
$A + B$	=	$B + A$
$A \cdot (B + C)$	=	$A \cdot B + A \cdot C$
$A \cdot (B \cdot C)$	=	$(A \cdot B) \cdot C$
$A + (B + C)$	=	$(A + B) + C$
$A + A \cdot B$	=	A
$A \cdot (A + B)$	=	A
$A \cdot (\overline{A} + B)$	=	$A \cdot B$
$A + \overline{A} \cdot B$	=	$A + B$
$\overline{A} + A \cdot B$	=	$\overline{A} + B$
$\overline{A} + A \cdot \overline{B}$	=	$\overline{A} + \overline{B}$

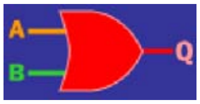
A AND B ≡ A . B
A OR B ≡ A + B

OR Operator (|)

fruits OR vegetables OR cereals
Any one of the terms are present.



A	B	A|B
0	0	0
0	1	1
1	0	1
1	1	1

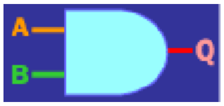


OR Gate

AND Operator (&)

dairy products AND export AND europe
All terms are present

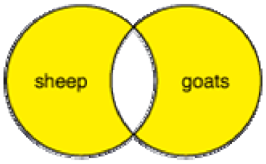
A	B	A&B
0	0	0
0	1	0
1	0	0
1	1	1



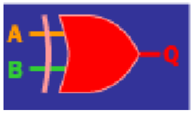
AND Gate

XOR Operator (Exclusive OR) (^)

sheep XOR goats
One or other term is present, but not both.



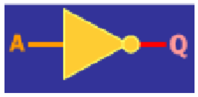
A	B	A^B
0	0	0
0	1	1
1	0	1
1	1	0



XOR Gate

NOT (~)

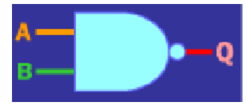
A	~A
0	1
1	0



NOT Gate

NAND

A	B	NAND
0	0	1
0	1	1
1	0	1
1	1	0

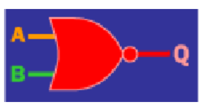


NAND Gate

All inputs on = off, else on.

NOR

A	B	NOR
0	0	1
0	1	0
1	0	0
1	1	0

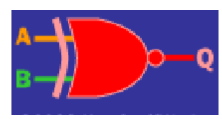


NOR Gate

Any input on = off, else on.

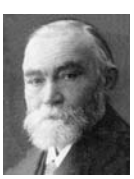
XNOR

A	B	NOR
0	0	1
0	1	0
1	0	0
1	1	1



XNOR Gate

FREGE: MATEMATICS AS LOGIC



Friedrich Ludwig Gottlob Frege

Born: 8 Nov 1848 in Wismar, Mecklenburg-Schwerin (now Germany)
Died: 26 July 1925 in Bad Kleinen, Germany

The Predicate Calculus (1)

- In an attempt to realize Leibniz's ideas for a language of thought and a rational calculus, Frege developed a formal notation (*Begriffsschrift*).
- He has developed the first predicate calculus: a formal system with two components: a formal language and a logic.

The Predicate Calculus (2)

The formal language Frege designed was capable of expressing:

- predicational statements of the form
  '*x* falls under the concept *F*' and '*x* bears relation *R* to *y*', etc.,
- complex statements such as
  'it is not the case that ...' and 'if ... then ...', and
- 'quantified' statements of the form
  'Some *x* is such that ...*x*...' and
  'Every *x* is such that ...*x*...'.

The Analysis of Atomic Sentences and Quantifier Phrases

Fred loves Annie.
Therefore, some *x* is such that *x* loves Annie.

Fred loves Annie.
Therefore, some *x* is such that Fred loves *x*.

Both inferences are instances of a single valid inference rule.

Proof

As part of his predicate calculus, Frege developed a strict definition of a '**proof**'.

In essence, he defined a *proof* to be any finite sequence of well-formed *statements* such that each statement in the sequence either is an *axiom* or follows from previous members by a valid *rule of inference*.

CANTOR: INFINITY

Georg Ferdinand Ludwig Philipp Cantor

Born: 3 March 1845 in St Petersburg, Russia
Died: 6 Jan 1918 in Halle, Germany

Infinities

Set of integers has an equal number of members as the set of even numbers, squares, cubes, and roots to equations!

The number of points in a line segment is equal to the number of points in an infinite line, a plane and all mathematical space!

The number of transcendental numbers, values such as π and *e* that can never be the solution to any algebraic equation, were much larger than the number of integers.

Hilbert described Cantor's work as:- ´...the finest product of mathematical genius and one of the supreme achievements of purely intellectual human activity.´

HILBERT: PROGRAM FOR MATHEMATICS

David Hilbert

Born: 23 Jan 1862 in Königsberg, Prussia (now Kaliningrad, Russia)
Died: 14 Feb 1943 in Göttingen, Germany

Hilbert's program

Provide a single formal system of computation capable of generating all of the true assertions of mathematics from “first principles” (first order logic and elementary set theory).

Prove mathematically that this system is consistent, that is, that it contains no contradiction. This is essentially a proof of correctness.

If successful, all mathematical questions could be established by mechanical computation!

GÖDEL: END OF HILBERTS PROGRAM

Kurt Gödel

Born: 28 April 1906 in Brünn, Austria-Hungary (now Brno, Czech Republic)
Died: 14 Jan 1978 in Princeton, New Jersey, USA

Incompleteness Theorems

1931 *Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme.*

In any axiomatic mathematical system there are propositions that cannot be proved or disproved within the axioms of the system.

In particular the consistency of the axioms cannot be proved.

TURING: UNIVERSAL AUTOMATON

Alan Mathison Turing

Born: 23 June 1912 in London, England
Died: 7 June 1954 in Wilmslow, Cheshire, England

When war was declared in 1939 Turing moved to work full-time at the Government Code and Cypher School at Bletchley Park.

Together with another mathematician W G Welchman, Turing developed **the Bombe**, a machine based on earlier work by Polish mathematicians, which from late 1940 was decoding all messages sent by the Enigma machines of the Luftwaffe.

At the end of the war Turing was invited by the National Physical Laboratory in London to design a computer.

His report proposing the Automatic Computing Engine (ACE) was submitted in March 1946.

Turing returned to Cambridge for the academic year 1947-48 where his interests ranged over topics far removed from computers or mathematics, in particular he studied *neurology* and *physiology*.

1948 Newman (professor of mathematics at the University of Manchester) offered Turing a readership there.

Work was beginning on the construction of a computing machine by F C Williams and T Kilburn. The expectation was that Turing would lead the mathematical side of the work, and for a few years he continued to work, first on the design of the subroutines out of which the larger programs for such a machine are built, and then, as this kind of work became standardized, on more general problems of numerical analysis.

1950 Turing published *Computing machinery and intelligence* in *Mind*

1951 elected a Fellow of the Royal Society of London mainly for his work on Turing machines

by 1951 working on the application of mathematical theory to biological forms.

1952 he published the first part of his theoretical study of morphogenesis, the development of pattern and form in living organisms.

VON NEUMAN: COMPUTER

John von Neumann

Born: 28 Dec 1903 in Budapest, Hungary
Died: 8 Feb 1957 in Washington D.C., USA

In the middle 30's, Neumann was fascinated by the problem of hydrodynamical turbulence.

The phenomena described by non-linear differential equations are baffling analytically and defy even qualitative insight by present methods.

Numerical work seemed to him the most promising way to obtain a feeling for the behaviour of such systems. *This impelled him to study new possibilities of computation on electronic machines ...*

Von Neumann was one of the pioneers of computer science making significant contributions to the development of logical design. Working in automata theory was a synthesis of his early interest in logic and proof theory and his later work, during World War II and after, on large scale electronic computers.

Involving a mixture of pure and applied mathematics as well as other sciences, automata theory was an ideal field for von Neumann's wide-ranging intellect. He brought to it many new insights and opened up at least two new directions of research.

He advanced the theory of cellular automata,
advocated the adoption of the bit as a measurement of computer memory, and
solved problems in obtaining reliable answers from unreliable computer components.

Computer Science Hall of Fame


Charles Babbage


Ada Countess of Lovelace


Axel Thue


Stephen Kleene


Julia Robinson


Noam Chomsky


Juris Hartmanis


John Brzozowski

Computer Science Hall of Fame


Richard Karp


Donald Knuth


Manuel Blum


Stephen Cook


Sheila Greibach


Leonid Levin

Women in Computer History

- Ada Byron King, Countess of Lovelace (1815-1852)
- Edith Clarke (1883-1959)
- Rósa Péter (1905-1977)
- Grace Murray Hopper (1906-1992)
- Alexandra Illmer Forsythe (1918-1980)
- Evelyn Boyd Granville
- Margaret R. Fox
- Erna Schneider Hoover
- Kay McNulty Mauchly Antonelli
- Alice Burks
- Adele Goldstine
- Joan Margaret Winters

Ada Byron King, Countess of Lovelace (1815-1852)

Ada heard in November, 1834, Babbage's ideas for a new calculating engine, the Analytical Engine. He conjectured: what if a calculating engine could not only foresee but could act on that foresight. Ada was touched by the "universality of his ideas". Hardly anyone else was.

In her article, published in 1843, Lady Lovelace's far-sighted comments included her predictions that such a machine might be used to compose complex music, to produce graphics, and would be used for both practical and scientific use.

Ada suggested to Babbage writing a plan for how the engine might calculate Bernoulli numbers. This plan, is now regarded as the first "computer program." A software language was named "Ada" in her honor in 1979.

Edith Clarke (1883-1959)

Edith Clarke is well-known in the field of Power Engineering. Her main contribution to the field was the development of tables that speeded up calculations for engineers. This was especially important because she created them during World War I, when engineers desperately needed to work faster.

Rósa Péter (1905-1977)

Her first research topic was number theory, but she became discouraged on finding that her results had already been proved by Dickson.

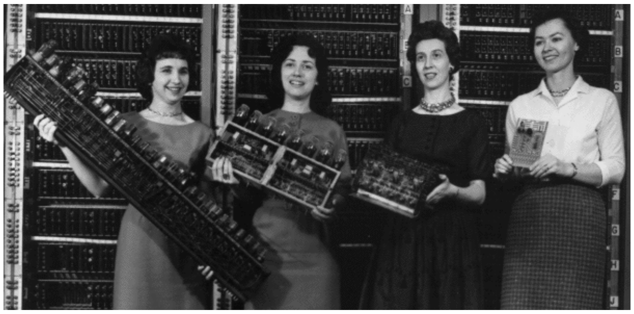
For a while Rósa wrote poetry, but around 1930 she was encouraged to return to mathematics by Kalmár. He suggested Rósa examine Gödel's work, and in a series of papers she became a founder of recursive function theory.

Rósa wrote Recursive Functions in 1951, which was the first book on the topic and became a standard reference.

In 1952 Kleene described Rósa Péter in a paper in *Bull. Amer. Math. Soc.* as ``the leading contributor to the special theory of recursive functions."

From the mid 1950's she applied recursive function theory to computers. In 1976 her last book was on this topic: *Recursive Functions in Computer Theory*.

ENIAC Ladies



Erna Schneider Hoover

Invention: Computerized Telephone Switching System

Erna Schneider earned a B.A. with honors in medieval history from Wellesley College, and later a Ph.D. in the philosophy and foundations of mathematics from Yale University.

In 1954, after teaching for a number of years at Swarthmore College, she began a research career at Bell Laboratories.

While there, she invented a computerized switching system for telephone traffic, to replace existing hard-wired, mechanical switching equipment. For this ground-breaking achievement -- the principles of which are still used today -- she was awarded one of the first software patents ever issued (Patent #3,623,007, Nov. 23, 1971) . At Bell Labs, she became the first female supervisor of a technical department.

Grace Murray Hopper (1906-1992)




- Grace Murray Hopper: **Inventor of the Computer Compiler**
- She participated in the development of the Common Business-Oriented Language (COBOL; 1959-61) for the UNIVAC
- The very **first computer bug**: Grace Murray Hopper originated this term when she found a real bug in a computer

Ida Rhodes (1900 -1986)

- She designed the C-10 language in the early 1950 for the UNIVAC.
- She also designed the original computer used for the Social Security Administration.
- In 1949, the department of Commerce awarded her an exceptional Service Gold Medal for significant pioneering leadership and outstanding contributions to the scientific progress of the nation.

Evelyn Boyd Granville

- **Evelyn Boyd Granville** - was one of the first African American women to earn a Ph.D. in Mathematics.
- She became a specialist in rocket and missile fuses, orbit computations and trajectory calculations for national defense and the space program providing technical support for the Vanguard, Mercury and Apollo projects. In addition, she served as an educational consultant to the State of California, helping to improve the teaching of math in elementary and secondary schools.

Jean E. Sammet

a preeminent pioneer in the computer world

She initiated the concept and directed the development of the first FORMAC (FORmula MAnipulation Compiler). FORMAC was the first widely used general language. It was also the first system for manipulating nonnumeric algebraic expressions.

In 1965, she became programming language technology manager in the IBM Systems Development Division. Afterward, she wrote a book on programming languages.

Her book, Programming Languages: History and Fundamentals, was published by Prentice-Hall in 1969.

Dr. Thelma Estrin
Professor Emerita
University of California,
Los Angeles

- Now retired from UCLA, where she was a computer science professor, Estrin was a pioneer in the field of biomedical engineering who realized that some of the most important ideas in science did not fit neatly into separate fields. Her work would combine concepts from anatomy, physiology, and neuroscience with electronic technology and electrical engineering. She was one of the first to use computer technology to solve problems in health care and in medical research.
- Estrin designed and then implemented the first system for analog-digital conversion of electrical activity from the nervous system," a precursor to the use of computers in medicine. She published papers on how to map the brain with the help of computers, and long before the Internet became popular and easy to use, she designed a computer network between UCLA and UC Davis in 1975.

Dana Angluin

B.A., Ph.D. University of California at Berkeley, 1969, 1976
Joined Yale Faculty 1979

- Algorithmic models of learning
- Professor Angluin's thesis was among the first work to apply complexity theory to the field of inductive inference.
- Her work on learning from positive data reversed a previous dismissal of that topic, and established a flourishing line of research. Her work on learning with queries established the models and the foundational results for learning with membership queries. Recently, her work has focused on the areas of coping with errors in the answers to queries, map-learning by mobile robots, and fundamental questions in modeling the interaction of a teacher and a learner.

Nancy A. Lynch

- **Professor of Electrical Engineering and Computer Science, Massachusetts Institute of Technology.** Nancy Lynch heads the *Theory of Distributed Systems Group (TDS) research group* in MIT's Laboratory for Computer Science. This group is part of the Theory of Computation (TOC) group and also of the Principles of Computer Systems (POCS) group.
- **Teaching** - Spring 2001: 6.897 Modeling and Analyzing Really Complicated Systems, Using State Machines
Fall 2001 6.852 Distributed Algorithms
- **Research interests**:
distributed computing, real-time computing, algorithms, lower bounds, formal modelling and verification.

Adele E. Goldberg

Cognitive Science

Adele Goldberg received her Ph.D. from the University of California at Berkeley in 1992. She is an associate professor in the UIUC Department of Linguistics and a part-time Beckman Institute faculty member in the Cognitive Science Group. Her fields of professional interest are syntax/semantics, constructional approaches to grammar, lexical semantics, language acquisition, language processing, and categorization.

Sandra L. Kurtzig

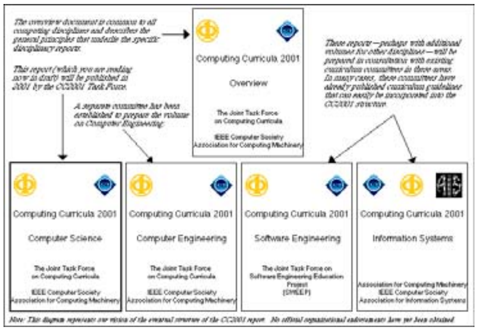
In today's male-dominated software industry, women founders and CEOs (chief executive officers) are practically nonexistent. But while software titans like Bill Gates and Oracle's Larry Ellison have become the poster boys for Silicon Valley success, the first multimillion-dollar software entrepreneur was a woman.

Starting with just $2,000, Sandra Kurtzig built a software empire that, at its peak, boasted around $450 million in annual sales. And it all started as a part-time job.

Internationale Unterschiede im prozentualen Frauenanteil am Informatik-Studium.

• England:	35% Frauen
• Italien, Frankreich, Spanien, Portugal	40- 50% Frauen
• Frühere Sowjetunion:	50% Frauen
• Bulgarien:	60-70% Frauen
• Griechenland:	59% Frauen
• Indien, Malaysia, Singapur:	50% Frauen
• Deutschland:	8% Frauen

COMPUTING
Overall structure of the CC2001 report



Technological advancement over the past decade

- The World Wide Web and its applications
- Networking technologies, particularly those based on TCP/IP
- Graphics and multimedia
- Embedded systems
- Relational databases
- Interoperability
- Object-oriented programming
- The use of sophisticated application programmer interfaces (APIs)
- Human-computer interaction
- Software safety
- Security and cryptography
- Application domains

Overview of the CS Body of Knowledge

- Discrete Structures
- Programming Fundamentals
- Algorithms and Complexity
- Programming Languages
- Architecture and Organization
- Operating Systems
- Net-Centric Computing
- Human-Computer Interaction
- Graphics and Visual Computing
- Intelligent Systems
- Information Management
- Software Engineering
- Social and Professional Issues
- Computational Science and Numerical Methods