Introduction to Logic
Conclusion

Michael Genesereth
Computer Science Department
Stanford University

Herbrand Logic

Multiple Logics
- Propositional Logic (logical operators)
- Relational Logic (variables and quantifiers)
- Functional Logic (functional terms)

Similarities and Differences
- Each language is a superset of its predecessor
- The semantics of each is a superset of its predecessor
- Each proof procedure is a superset of its predecessor

We can think of them all as a single logic.
*Together called **Herbrand Logic**.*

Language of Herbrand Logic

$$p(a) \qquad q(b,c)$$

$$\neg p(b) \qquad \forall x.(p(x) \Rightarrow q(x,f(x)))$$

$$p(c) \vee p(d) \qquad \exists x.q(x,d)$$

Herbrand Semantics

The *Herbrand base* is the set of all ground atoms.

$$\{p(a), p(b), q(a,a), q(a,b), q(b,a), q(b,b)\}$$

A *Herbrand model* is an arbitrary subset we take as true.

$$\{p(a), q(a,b), q(b,a)\}$$

Equivalently, a Herbrand model is a *truth assignment*.

$p(a)=1$	$q(a,a)=0$	$q(b,a)=1$
$p(b)=0$	$q(a,b)=1$	$q(b,b)=0$

Given a truth assignment, the semantics of our language gives us a truth assignment for all sentences.

$$(\forall x.(p(x) \Rightarrow \neg q(x, x))) = 1$$

Properties of Sentences

Valid	A sentence is *valid* if and only if *every* interpretation satisfies it.
Contingent	A sentence is *contingent* if and only if *some* interpretation satisfies it and *some* interpretation falsifies it.
Unsatisfiable	A sentence is *unsatisfiable* if and only if *no* interpretation satisfies it.

Relationships Between Sentences

A sentence ϕ is *logically equivalent* to a sentence ψ if and only if every truth assignment that satisfies ϕ satisfies ψ *and* every truth assignment that satisfies ψ satisfies ϕ.

A premise φ *logically entails* a conclusion ψ if and only if every interpretation that satisfies φ also satisfies ψ.

A sentence ϕ is *consistent with* a sentence ψ if and only if there is a truth assignment that satisfies both ϕ and ψ.

Semantic Reasoning

Logical Entailment is effectively a comparison of truth tables, logic grids, etc.

Constants		Premises
p	**q**	**p & q**
1	1	1
1	0	0
0	1	0
0	0	0

Constants		Premises
p	**q**	**p | q**
1	1	1
1	0	1
0	1	1
0	0	0

NB: Truth tables and logic grids can be very large or even infinite, rendering this approach impractical in general.

Proof

A *proof* of a conclusion from a set of premises is a sequence of sentences terminating in the conclusion in which each item is either (1) a premise, (2) an instance of an axiom schema, or (3) the result of applying a rule of inference to earlier items in sequence.

1.	p	Premise
2.	$p \Rightarrow q$	Premise
3.	$(p \Rightarrow q) \Rightarrow (q \Rightarrow r)$	Premise
4.	q	IE: 2, 1
5.	$q \Rightarrow r$	IE: 3, 2
6.	r	IE: 5, 4

Soundness and Completeness

A proof system is *sound* if and only if every provable conclusion is logically entailed.

$$\text{If } \Delta \vdash \phi, \text{ then } \Delta \models \phi.$$

A proof system is *complete* if and only if every logical conclusion is provable.

$$\text{If } \Delta \models \phi, \text{ then } \Delta \vdash \phi.$$

Results

Propositional Logic

Logical entailment and provability are identical

Logical entailment is decidable

Relational Logic

Logical entailment and provability are identical

Logical entailment is decidable

Functional Logic

More expressive, more powerful, *but*...

Some logically entailed results have only *infinite proofs*

There is *no* sound and complete proof procedure

Logical entailment is *not* decidable

What's Next?

Metalevel Representation

Sentences as arguments in other sentences.

Belief and knowledge

$$\forall x.(believes(john, p(x)) \wedge automobile(x) \Rightarrow p(x))$$

$$\forall x.(believes(john, p(x)) \wedge believes(mary, p(x)) \Rightarrow p(x))$$

$$\forall x.(p(x) \Rightarrow believes(john, p(x)))$$

Higher Order Logic

Functional and Relational Variables

Properties of Relations

$\forall r.(reflexive(r) \Leftrightarrow \forall x{:}(r(x,x))$

$\forall r.(symmetric(r) \Leftrightarrow \forall x{:}\forall y{:}(r(x,y) \Rightarrow r(y,x)))$

$\forall r.(transitive(r) \Leftrightarrow \forall x{:}\forall y{:}\forall z{:}(r(x,y)\ \&\ r(y,z) \Rightarrow r(x,z)))$

Minimization:

$\forall r.(subrelation(q,r) \Leftrightarrow \forall x{:}\forall y{:}(q(x,y) \Rightarrow r(x,y)))$

$\forall r.(proper(p,q) \Leftrightarrow subrelation(p,q)\ \&\ {\sim}subrelation(q,p))$

$\forall r.(transitivecover(p,r) \Leftrightarrow subrelation(p,r)\ \&\ transitive(r))$

$\forall r.(transitiveclosure(p,r) \Leftrightarrow$

$transitivecover(p,r) \wedge$

$\neg\exists q{:}(transitivecover(p,q) \wedge proper(q,r)))$

Deduction

A premise φ *logically entails* a conclusion ψ if and only if every interpretation that satisfies φ also satisfies ψ.

Deduction is sound reasoning. If the premises are true, the conclusion *must* be true (i.e. logically entailed).

Incomplete Induction

Reasoning from the specific to the general

I have seen 1000 black ravens.
I have never seen a raven that is not black.
Therefore, every raven is black.

Until you see an exception.

Induction is the basis for Science (and machine learning)
Deduction is the subject matter of Logic.

Science aspires to discover new knowledge.
Logic aspires to derive conclusions implied by premises.

Abduction

Reasoning to premises from conclusions / observations

If there is no fuel, the car will not start.
If there is no spark, the car will not start.
There is spark.
The car will not start.
Therefore, there is no fuel.

What if the car is in a vacuum chamber?

Often used in diagnostic reasoning

Analogy

Reasoning based similarity of situations

The flow in a pipe is proportional to its diameter.
Wires are like pipes.
Therefore, the current in a wire is proportional to diameter.

Now try price.

Often used in argumentation

Default Reasoning

Reasoning based on absence of contradictory evidence.

General Pattern:

If p(a) not known to be true, assume it is false.

Modular Arithmetic:

$same(0,0)$	$\neg same(1,0)$	$\neg same(2,0)$	$\neg same(3,0)$
$\neg same(0,1)$	$same(1,1)$	$\neg same(2,1)$	$\neg same(3,1)$
$\neg same(0,2)$	$\neg same(1,2)$	$same(2,2)$	$\neg same(3,2)$
$\neg same(0,3)$	$\neg same(1,3)$	$\neg same(2,3)$	$same(3,3)$

Problem Case:

$p(a) \vee p(b)$

We do not know that $p(a)$ is true.

Do we infer $\neg p(a)$? Do we infer $\neg p(b)$?

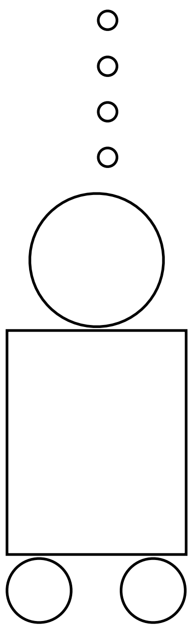
Automated Reasoning

Computational Logic

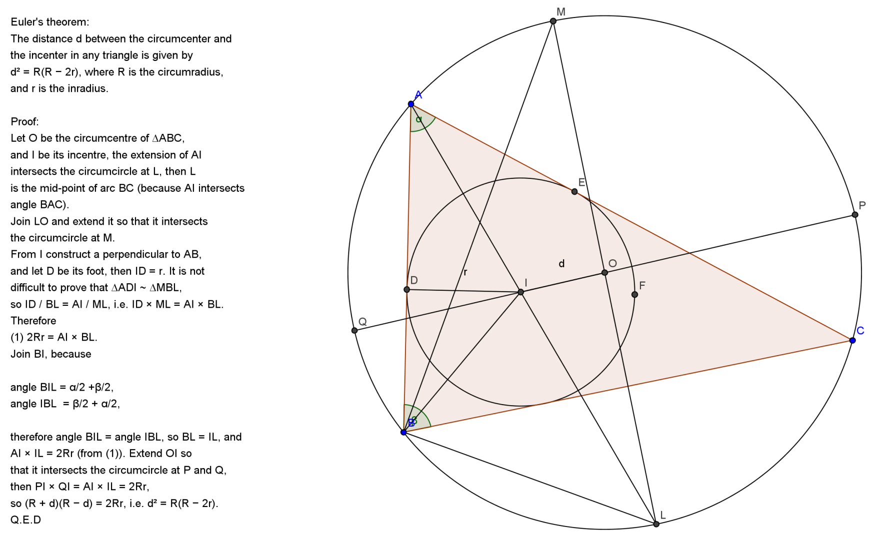
$p(a,b)$ $\qquad$ $q(b,c)$

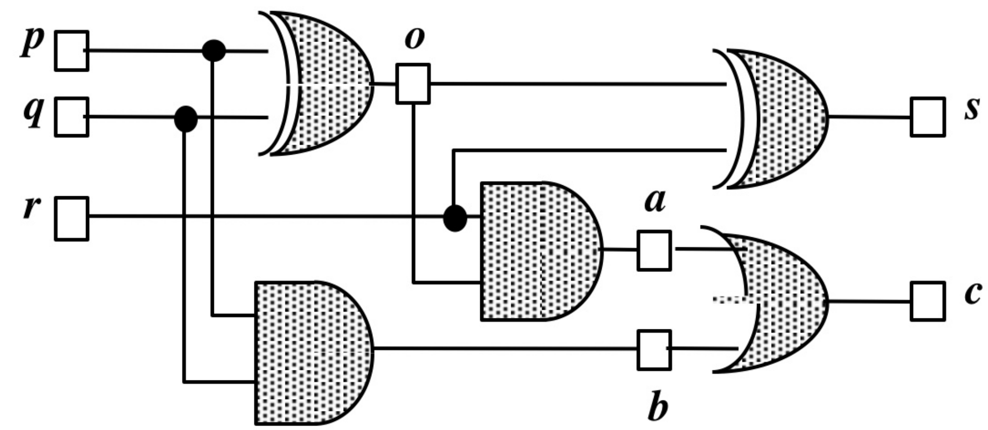
$\neg p(b,d)$ $\qquad$ $\forall x.\forall y.(p(x,y) \Rightarrow q(x,y))$

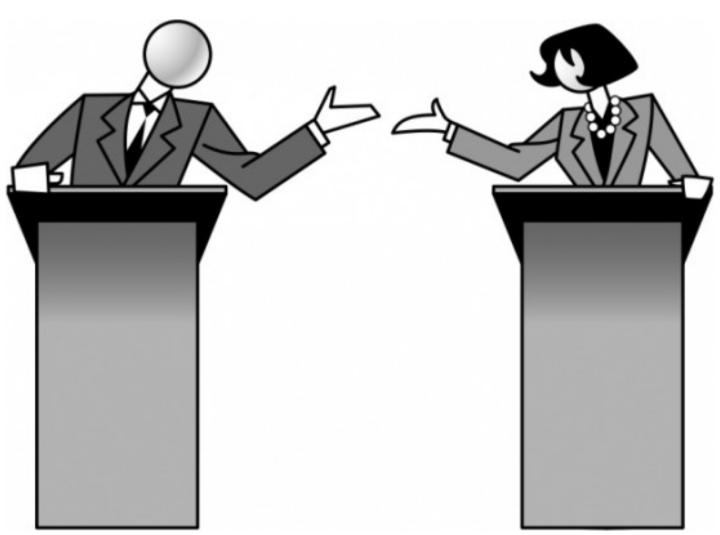
$p(c,b) \vee p(c,d)$ $\qquad$ $\exists x.p(x,d)$



Applications







Automated Theorem Proving

Chang and Lee: Mechanical Theorem Proving
https://doi.org/10.1016/c2009-0-22103-9

Larry Wos: Automated Reasoning Website
http://www.automatedreasoning.net

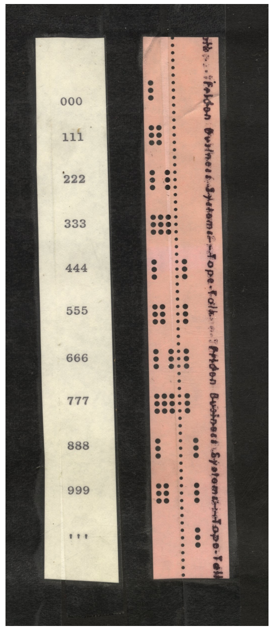
Prolog Technology Theorem Prover
http://www.ai.sri.com/~stickel/pttp.html

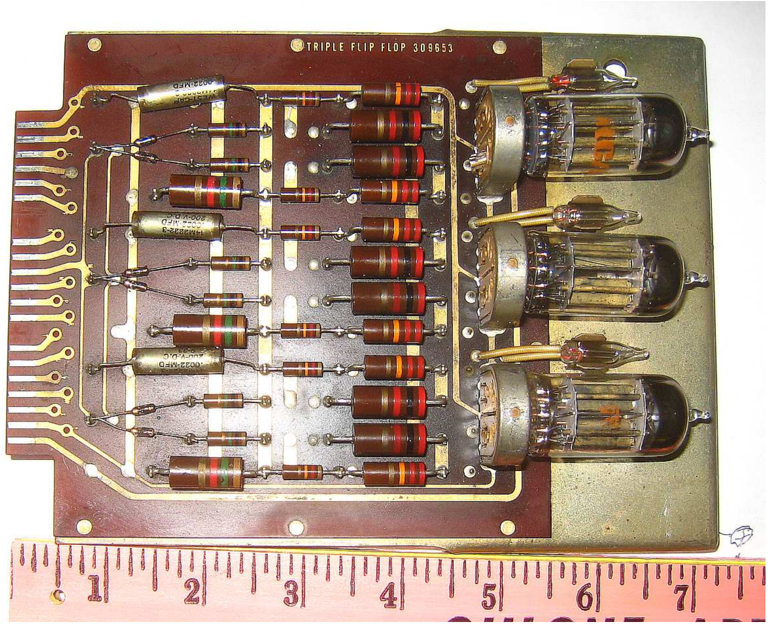
Prover9 / Mace4
https://www.cs.unm.edu/~mccune/prover9/

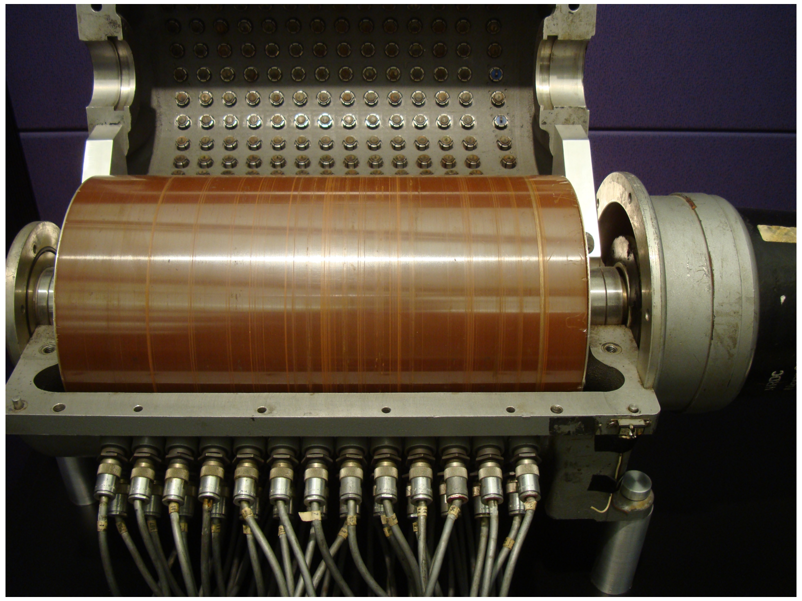
Thousands of Problems for Theorem Provers
http://www.tptp.org

Logic Programming

LGP-30


000
111
222
333
444
555
666
777
888
999


TRIPLE FLIP FLOP 309653
1
2
3
4
5
6
7



IBM 360

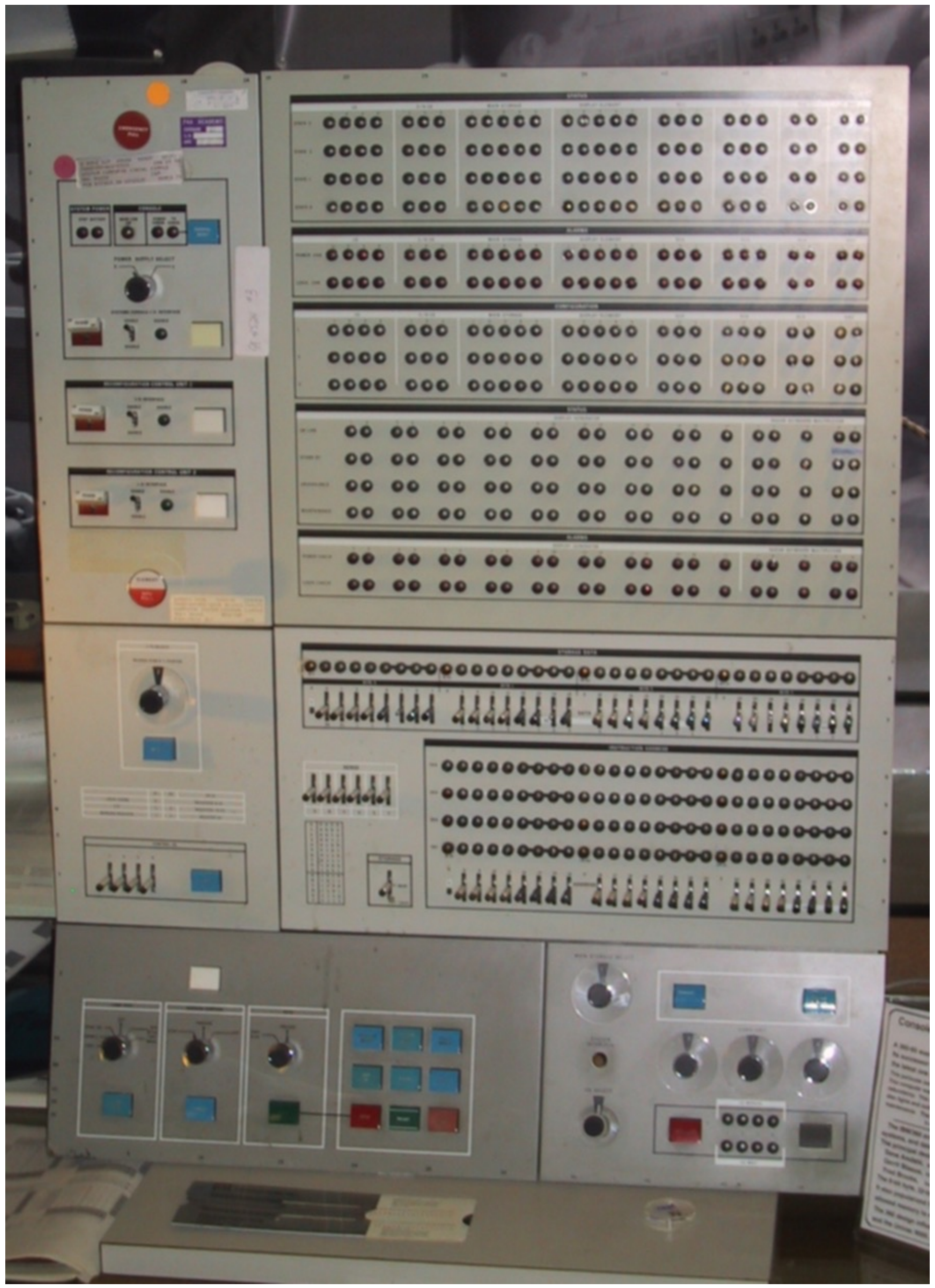
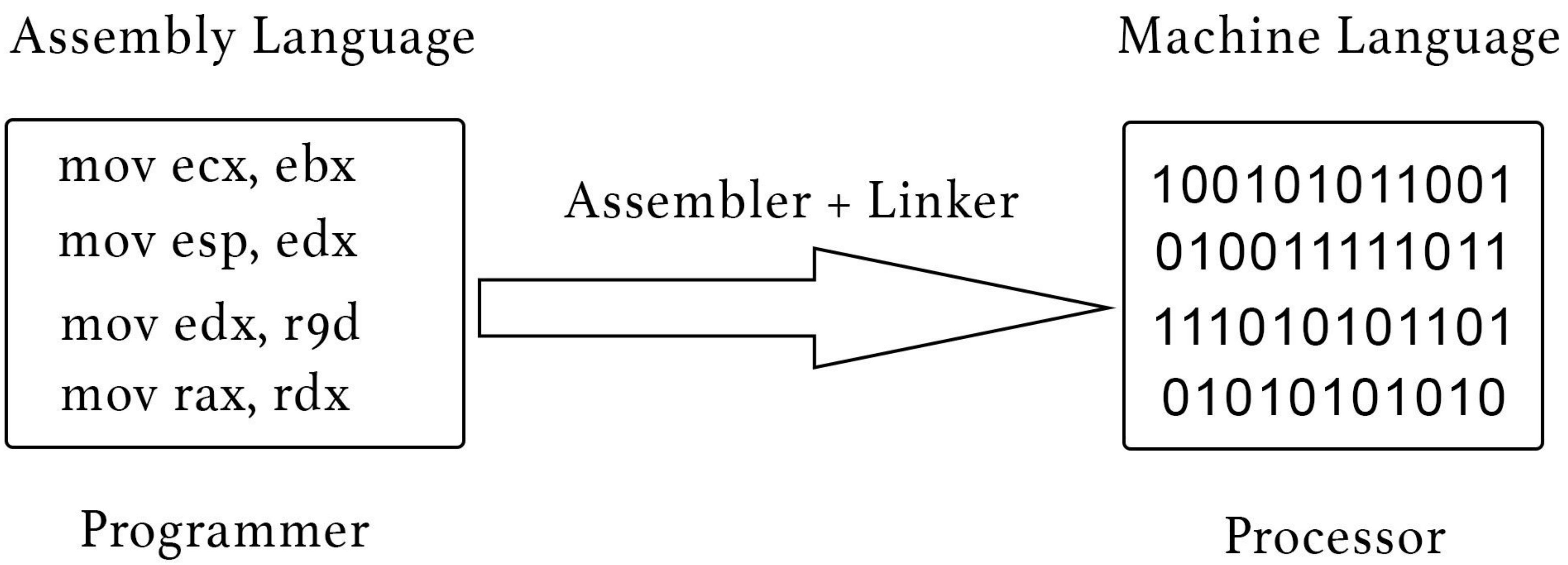
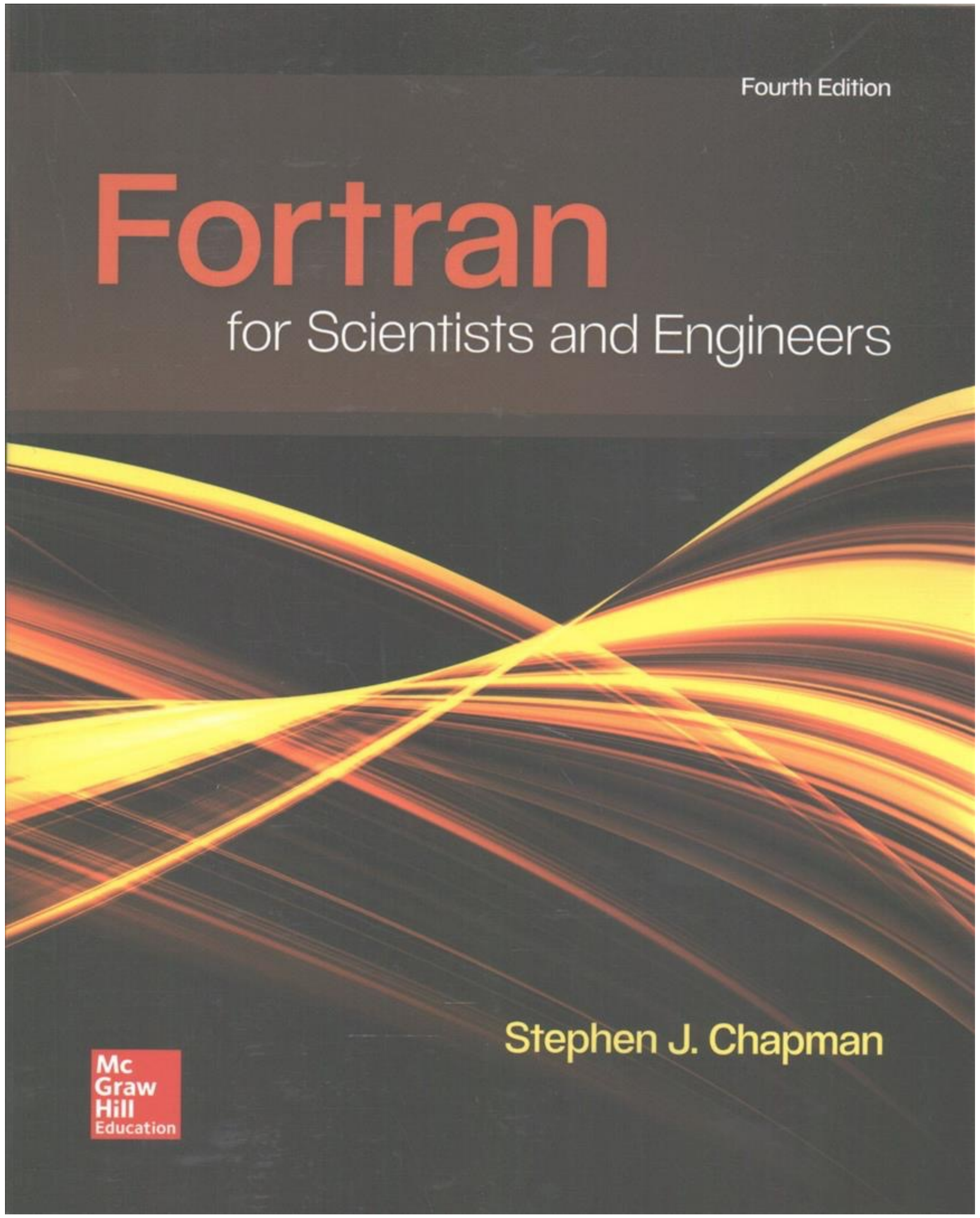


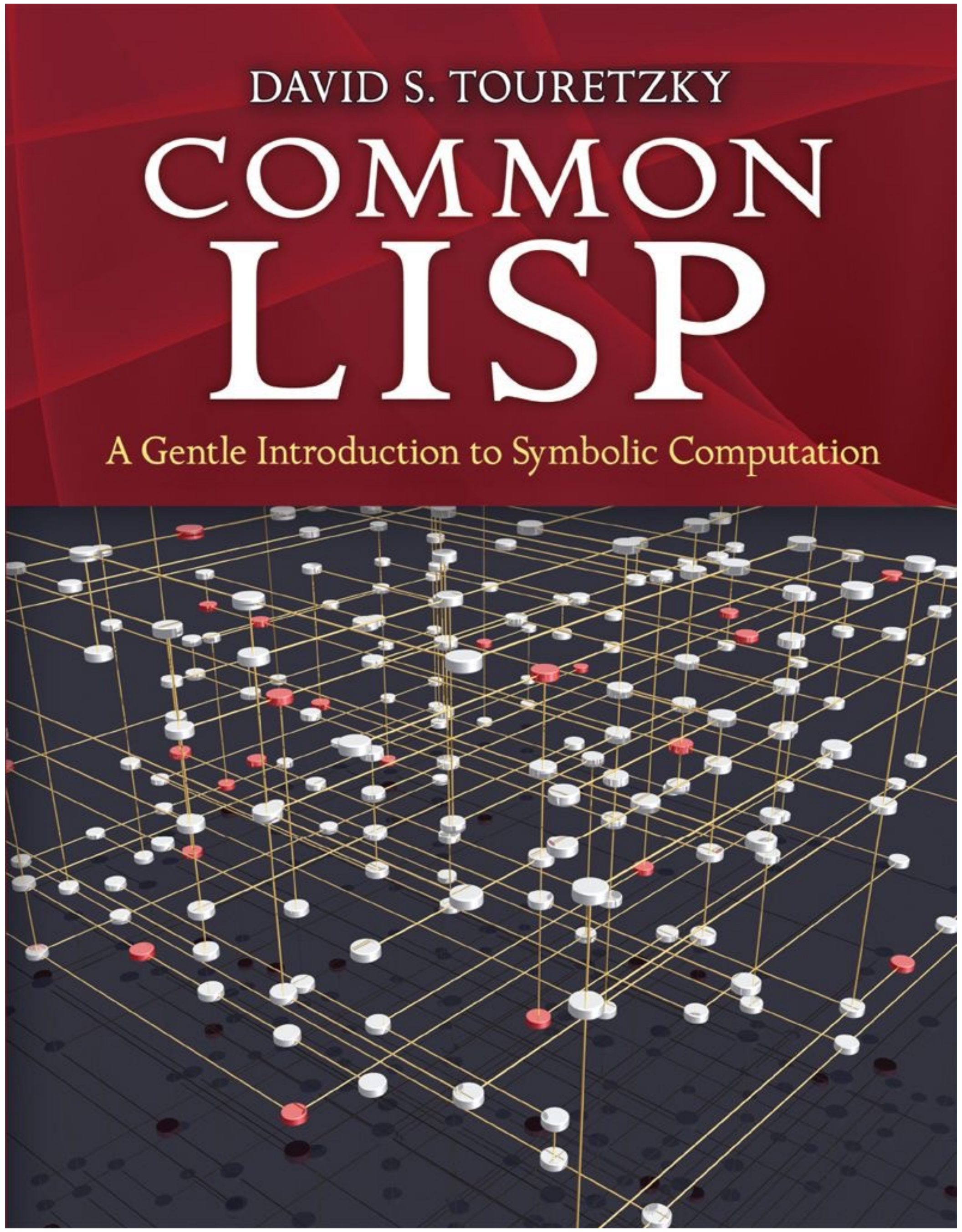
Figure 4. Card Codes and Graphics for 64-Character Set

Assembly Language

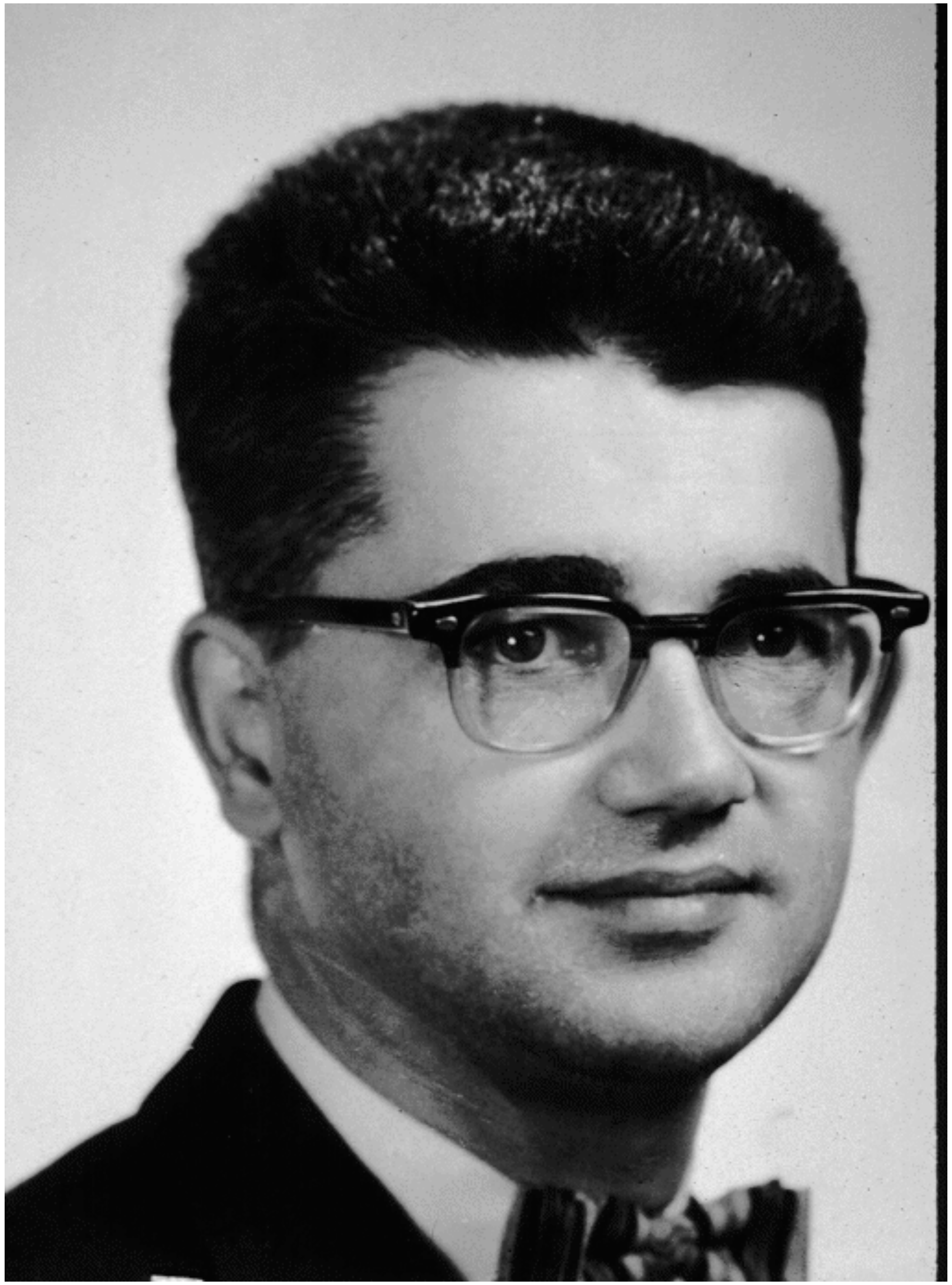


Higher Level Languages





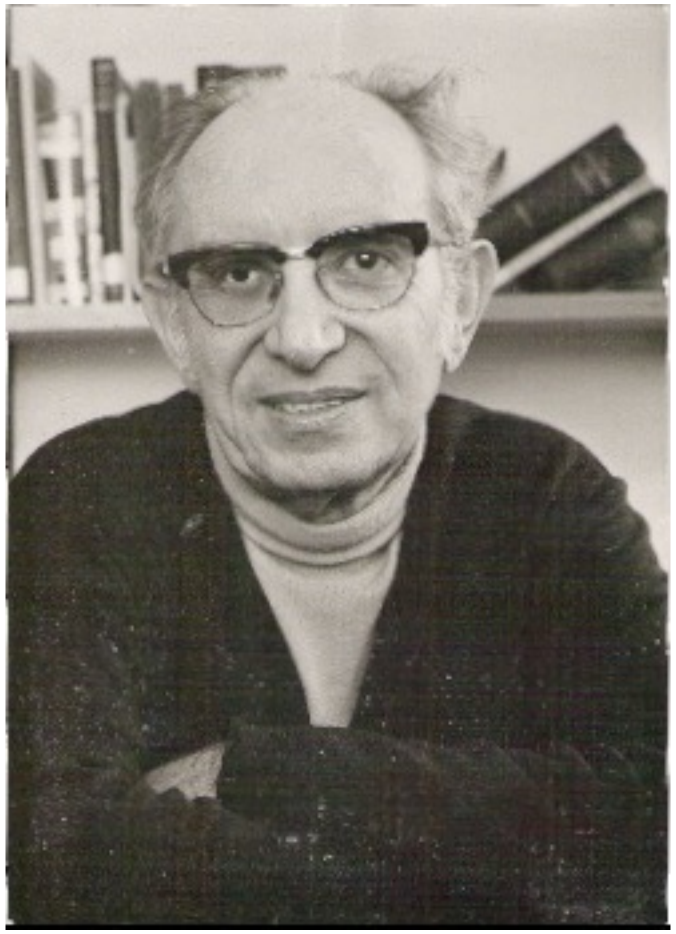
John McCarthy



*The main advantage we expect the **advice taker** to have is that its behavior will be improvable merely by making statements to it, telling it about its … environment and what is wanted from it.*

- John McCarthy1958

Bar-Hillel



McCarthy's paper belongs in the Journal of Half-Baked Ideas ... The gap between McCarthy's general programme and its execution ... seems to me so enormous that much more has to be done to persuade me that even the first step in bridging the gap has already been taken.

- Yehoshua Bar-Hillel 1958

Ed Feigenbaum

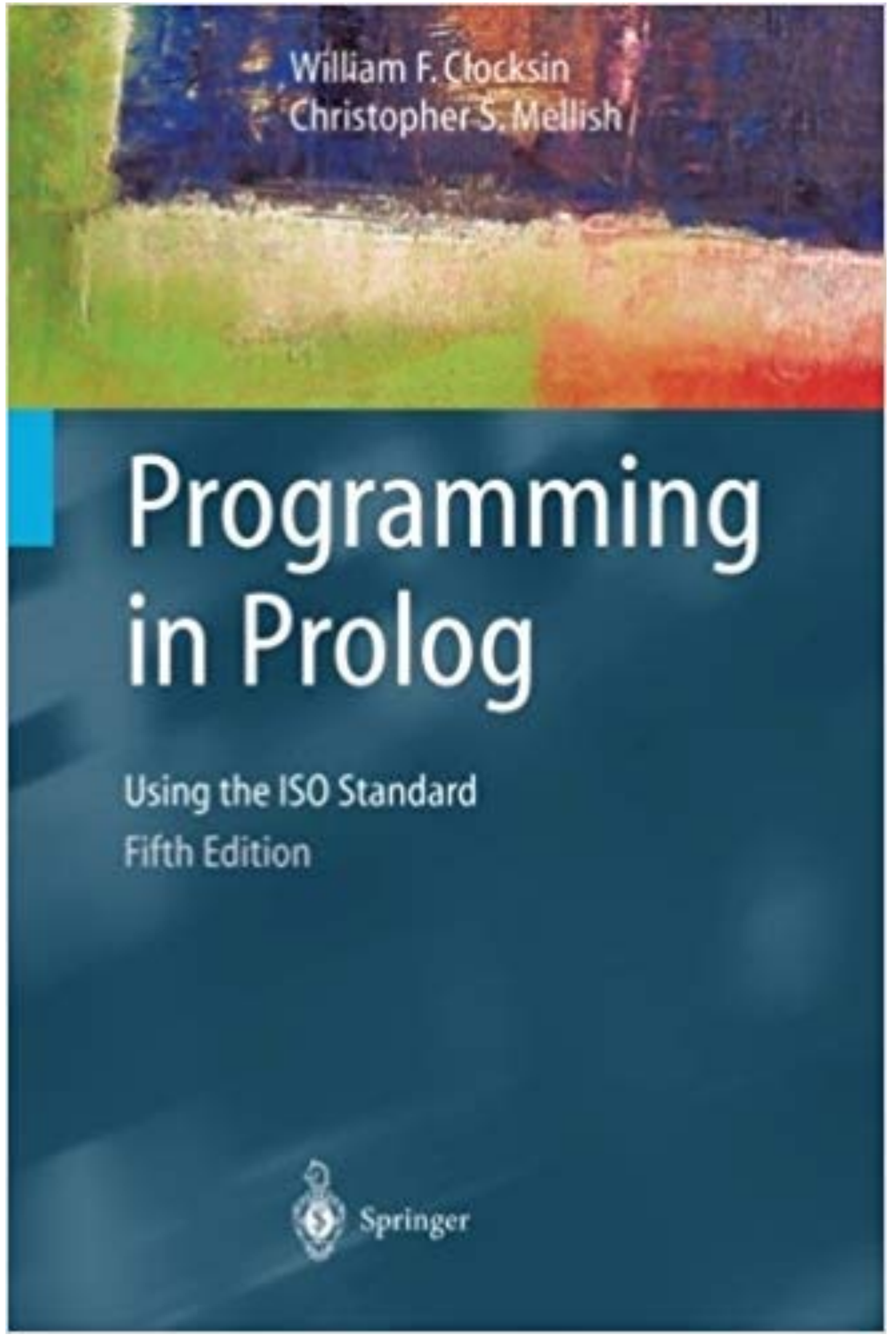
The potential use of computers by people to accomplish tasks can be "one-dimensionalized" into a spectrum representing the nature of the instruction that must be given the computer to do its job. Call it the ***what-to-how spectrum****. At one extreme of the spectrum, the user supplies his intelligence to instruct the machine with precision exactly how to do his job step-by-step. ... At the other end of the spectrum is the user with his real problem. ... He aspires to communicate what he wants done ... without having to lay out in detail all necessary subgoals for adequate performance.*

- Ed Feigenbaum 1974

Databases

SQL

Bob Kowalski



http://cs151.stanford.edu