Information in the Philosophy of Computer Science

Giuseppe Primiero

UNIVERSITÀ DEGLI STUDI DI MILANO

DIPARTIMENTO DI FILOSOFIA
"PIERO MARTINETTI"

The Central Role of Information for CS

Information plays a central role in computer science. The discipline is hardly comprehensible when abstracted from the conceptualization and use of this notion.

Computer Science is..

"the art and science of representing and processing information" [Forsythe, 1967]

"the study of information structures and processes and how [they] can be implemented on a digital computer" [Atchison et al., 1968]

"the study of representation, transformation, and interpretation of information structures" [Wegner, 1976]

"the body of knowledge of information-transforming processes" [Denning, 1985]

"the study of information" [Hartmanis and Lin, 1992]

"information engineering" [Bajcsy and Reynolds, 2002]

"a science that studies naturally and artificially occurring information processes" [Denning, 2007]

Information and the Discipline

Information is present through all subfields of the discipline:

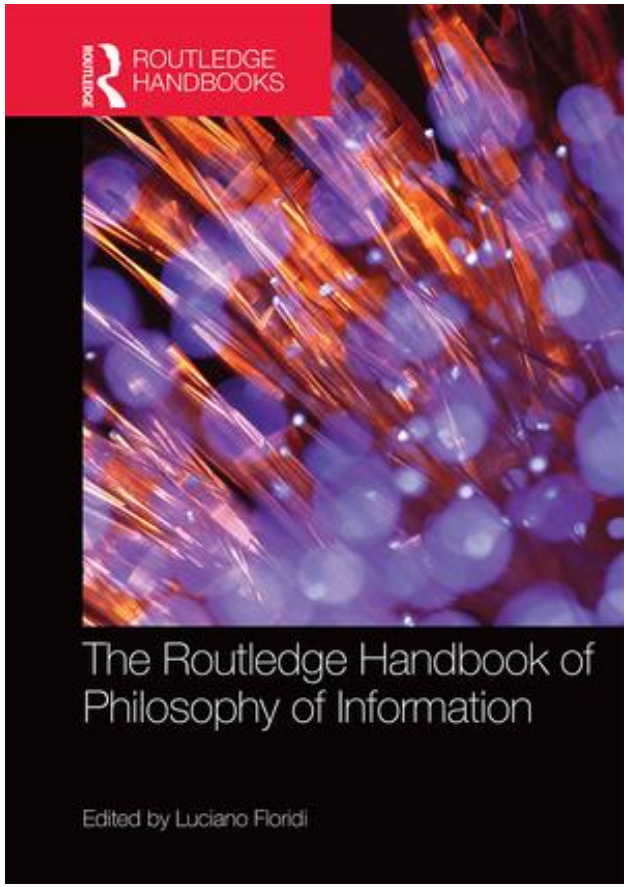
- formal structures of data and algorithms
- language implementation
- program design
- hardware manipulation
- data visualization
- data analytics, processing and machine learning
- autonomous agents
- bioinformatics

Information Types

Conceptually, different types of information are at work for each sub-discipline:

- syntactic information
- semantic information
- procedural information
- abstract information
- intentional information

Our task, [Primiero, 2016]



- Analyse information at each LoAs of the computational model;
- Reveal a structure based on
  1. an ontology of *syntax-semantics*
  2. an epistemology of *control*

Information Inside the Computing Machine: Structured Data

Operational Information: Controlling Structured Data

Instructional Information: Programs and their Semantics

Abstract Information: Algorithm, Design and Purpose

Ontology and Epistemology

Information Inside the Computing Machine: Structured Data

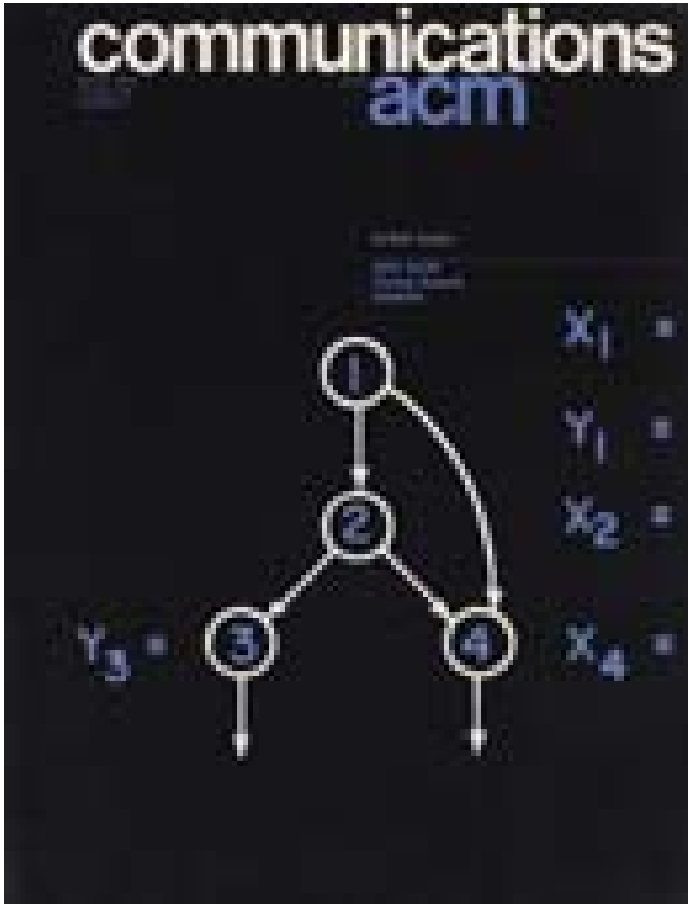
Operational Information: Controlling Structured Data

Instructional Information: Programs and their Semantics

Abstract Information: Algorithm, Design and Purpose

Ontology and Epistemology

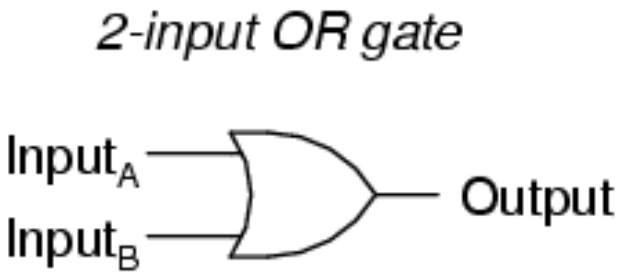
The mechanical core



"the science of computers and related phenomena" [Newell et al., 1967]

"the empirical study of computer related phenomena" [Newell and Simon, 1976]

Information in the mechanical core



A	B	Output
0	0	0
0	1	1
1	0	1
1	1	1

- The core business of computer science is the material execution and mechanical realization of information-transforming processes [Denning, 1985]
- Information is binary digits (bits) expressing discrete, exclusive ON/OFF states of electrical input.

Structured Data

- Ontology: electrical input is the ontological domain
- Structure:
  1. value assignment: (possibly compound) $x := 0 \mid 1$;
  2. value dependency: output y whose value depends on x;
  3. rule execution: dependency between x and y is rule based.

Information as Structured Data

Definition (Structured Data)

Information is physically evaluated variables, with structure control associating electrical charges to actions.

Building the Control structure

- This notion of information tells us what happens through data structuring
- To explain *how* is this controlled, we need to go to a higher level of abstraction.

Information Inside the Computing Machine: Structured Data

Operational Information: Controlling Structured Data

Instructional Information: Programs and their Semantics

Abstract Information: Algorithm, Design and Purpose

Ontology and Epistemology

The process of structuring information

- technically: low-level languages control information flow for the appropriate architecture;
- conceptually: it requires defining actions (knowledge-that) in terms of operations (knowledge-how).

Example

```
DATA SEGMENT
NUM DW 1234H, 0F234H
SUM DW 2 DUP(0)
DATA ENDS
CODE SEGMENT
ASSUME CS: CODE, DS:DATA
START: MOV AX,DATA
MOV DS,AX
MOV AX,NUM ; First number loaded into AX
MOV BX,0H ; For carry BX register is cleared
ADD AX,NUM+2 ; Second number added with AX
JNC DOWN ; Check for carry
INC BX ; If carry generated increment the BX
DOWN: MOV SUM,AX ; Storing the sum value
MOV SUM+2,BX ; Storing the carry value
MOV AH,4CH
INT 21H
CODE ENDS
END START
```

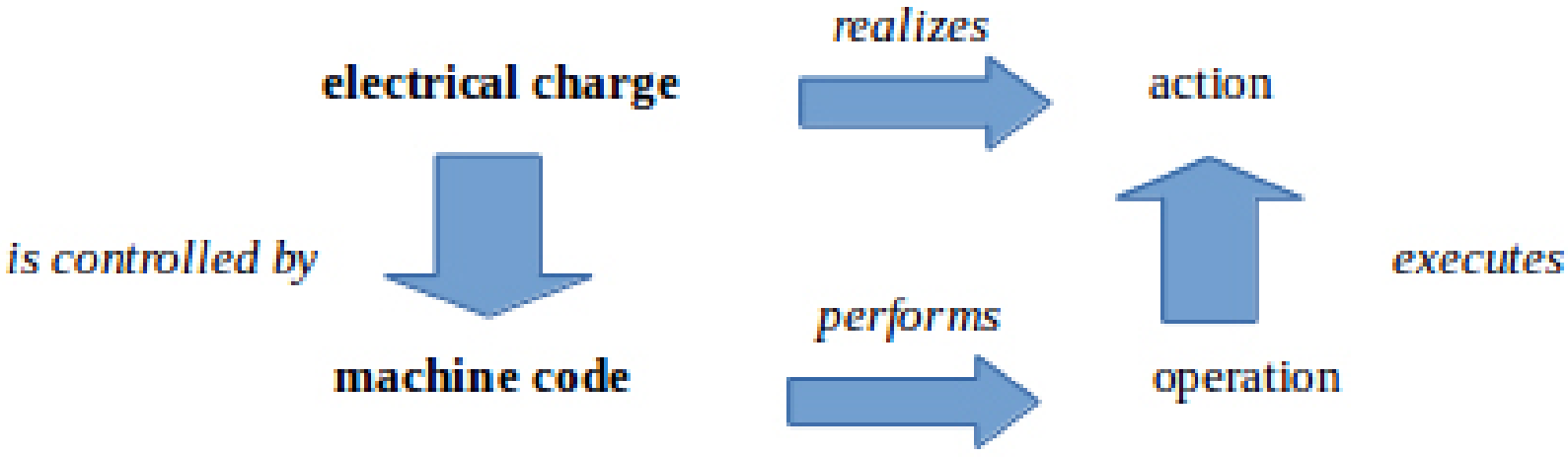
Ontology of Structure: Operational Information

Definition (Operational Information)

Information is

- syntactically well-defined data
- denoting the ontology of the physical layer, fixed by the architecure

New Control Structure



Information Inside the Computing Machine: Structured Data

Operational Information: Controlling Structured Data

Instructional Information: Programs and their Semantics

Abstract Information: Algorithm, Design and Purpose

Ontology and Epistemology

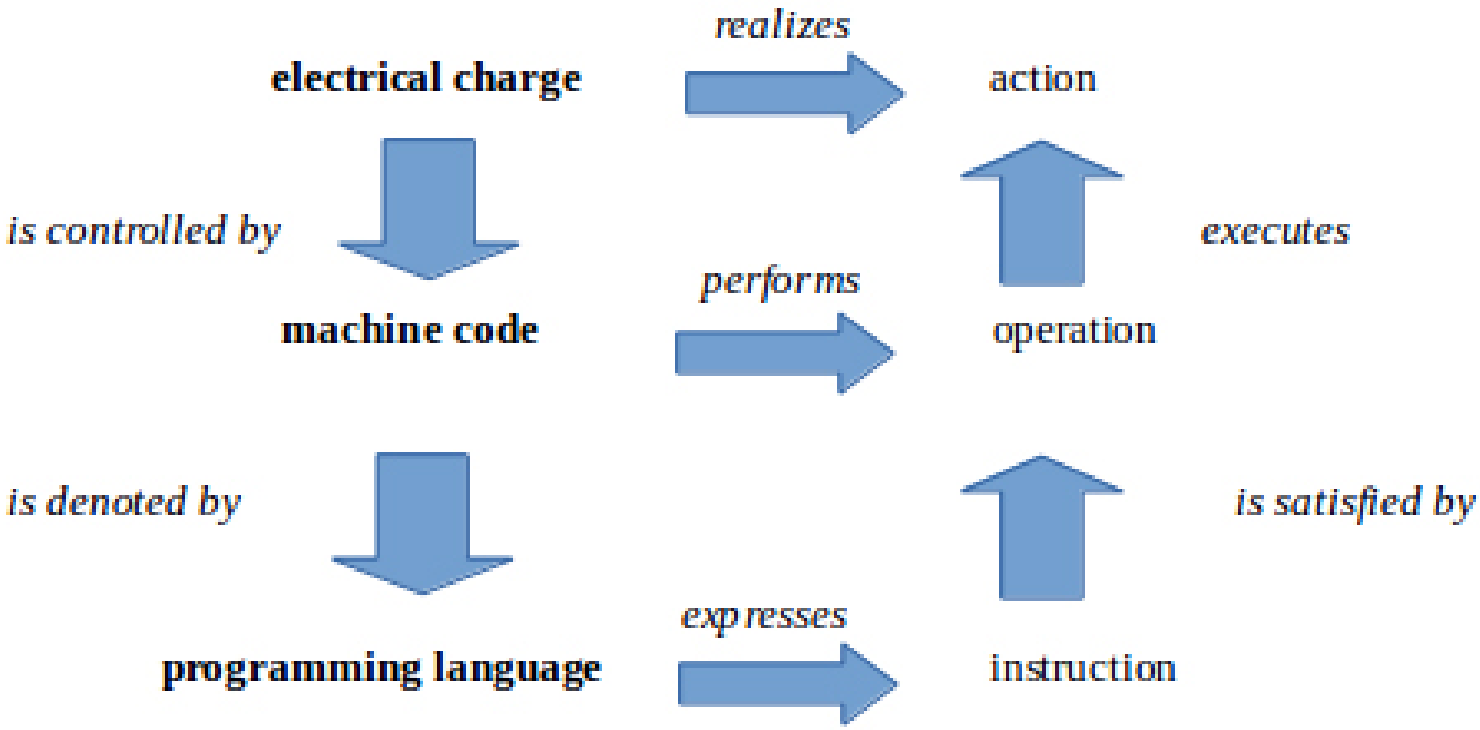
Instructions

- Abstraction from physical layer
- Instructions: interface between the user and the machine language by means of a programming language.

Example

```
int main() {
int a, b, c;
printf("Enter two numbers to add\n");
scanf("%d%d",&a,&b);
c = a + b;
printf("Sum of entered numbers = %d\n",c);
return 0;
}
```

New control structure



Information of Instructions

Definition (Instructional Information)

Instructional information has

- an ontology of abstract objects and their properties (PL constructs and rules);
- no alethic evaluation
- correct in view of implementation: is the execution as intended?

Information Inside the Computing Machine: Structured Data

Operational Information: Controlling Structured Data

Instructional Information: Programs and their Semantics

Abstract Information: Algorithm, Design and Purpose

Ontology and Epistemology

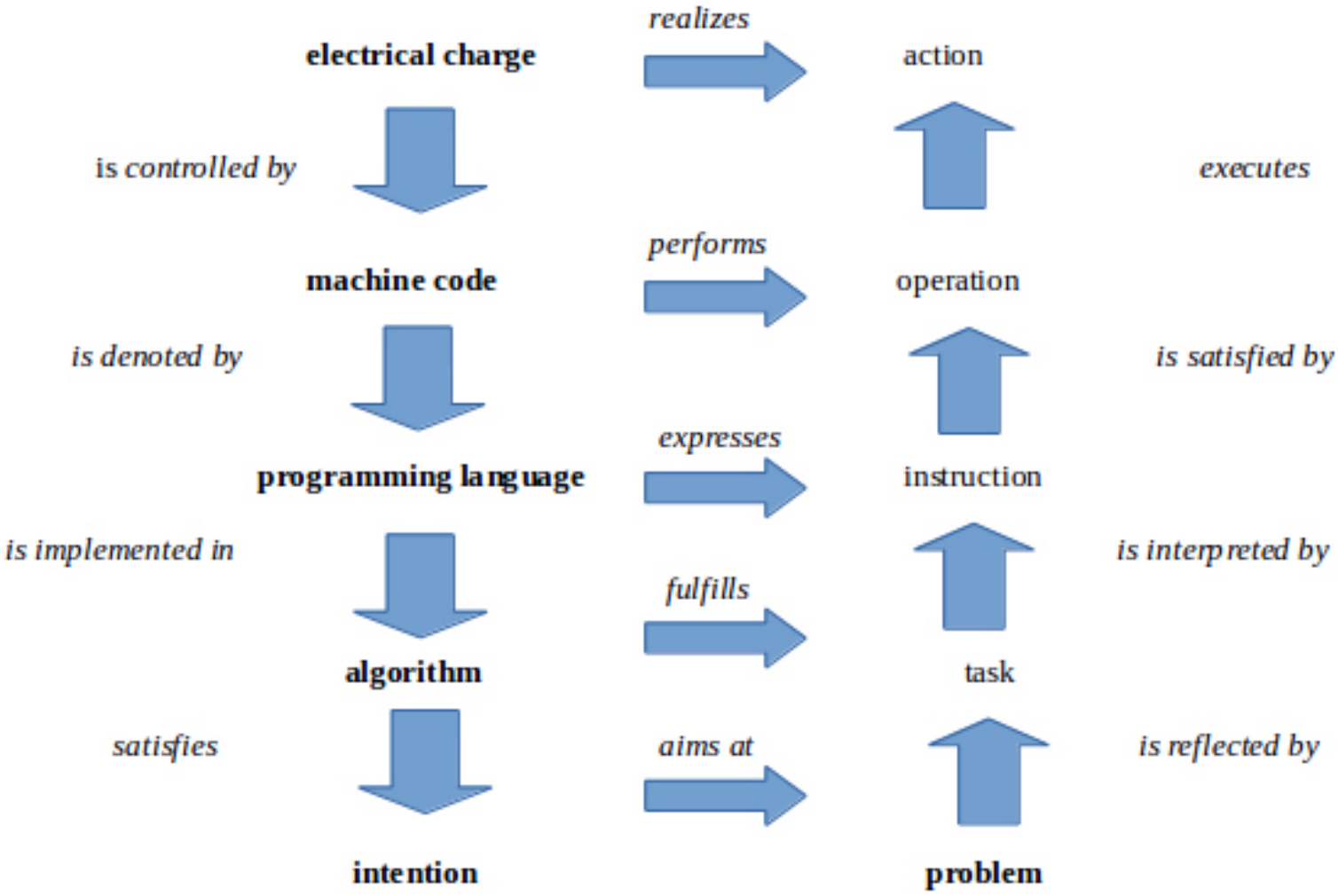
Commands

- A program is designed to implement an algorithm.
- An algorithm is the abstract representation of a mathematical function
- A mathematical function is required to fulfill a task.

Example

- read the Values of A and B
- if A and B != 0, SUM = A+B. Display SUM. Stop.
- Otherwise, Return ERROR: ‘No positive inputs’. Stop.

New control structure



Information of Algorithms

Definition (Abstract Information)

Information in algorithms is

- abstract,
- it determines the correctness of any of its implementations
- it has a true the domain of functions, i.e. all the functions satisfied by it.

Information Inside the Computing Machine: Structured Data

Operational Information: Controlling Structured Data

Instructional Information: Programs and their Semantics

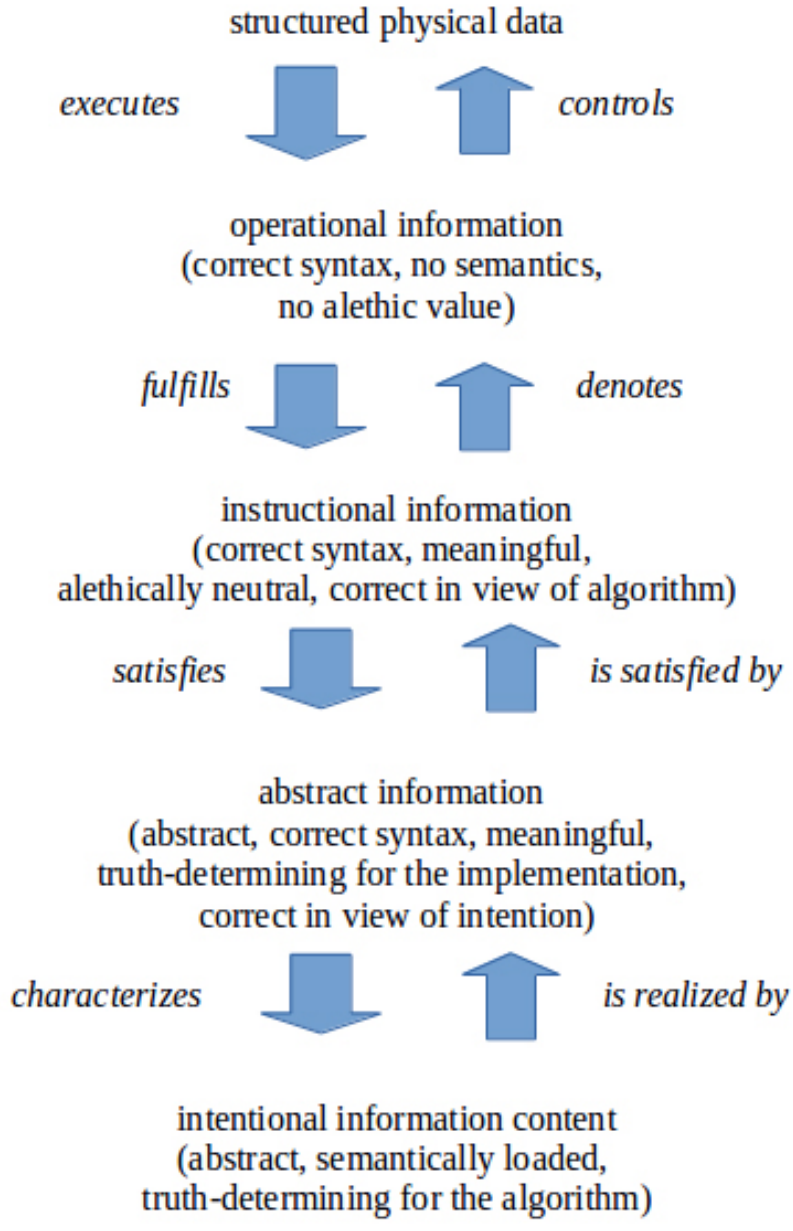
Abstract Information: Algorithm, Design and Purpose

Ontology and Epistemology

Ontology and Epistemology

- Ontology of Computation: the relation abstraction-implementation, realized in the syntax-semantics levels;
- Epistemology of Computation: the control structure that the ontology instantiates.

Information Flow



Ontology and Epistemology

Epistemological Construct		Ontological Domain
Problem	aimed by	**Intention**
Task	fulfilled by	**Algorithm**
Instruction	expressed by	**Programming Language**
Operation	performed by	**Machine Code**
Action	realised by	**Electrical Charge**

Figure: Elements of the Epistemology-Ontology pairs.

A Science of Information

Definition (Computing as Science of Information)

Computing is the systematic study of the ontologies and epistemology of information structures.

Computing, today more than ever before, is a multi-faceted discipline which collates several methodologies, areas of interest, and approaches: mathematics, engineering, programming, applications. Given its enormous impact on everyday life, it is essential that its debated origins are understood, and that its different foundations are explained. *On the Foundations of Computing* offers a comprehensive and critical overview of the birth and evolution of computing, and it presents some of the most important technical results and philosophical problems of the discipline, combining both historical and systematic analyses.

The debates this text surveys are among the latest and most urgent ones: the crisis of foundations in mathematics and the birth of the decision problem, the nature of algorithms, the debates on computational artefacts and malfunctioning, and the analysis of computational experiments. By covering these topics, *On the Foundations of Computing* provides a much-needed resource to contextualize these foundational issues.

For practitioners, researchers, and students alike, a historical and philosophical approach such as what this volume offers becomes essential to understand the past of the discipline and to figure out the challenges of its future.

Giuseppe Primiero is Associate Professor of Logic at the Department of Philosophy, University of Milan, Italy.

Cover image: Blue Andy/Shutterstock.com

ISBN 978-0-19-883565-3

9 780198 835653

OXFORD
UNIVERSITY PRESS

www.oup.com

Primiero

ON THE FOUNDATIONS OF COMPUTING

OXFORD

ON THE FOUNDATIONS OF COMPUTING

Giuseppe Primiero

OXFORD

Thanks

References I


Atchison, W. F., Conte, S. D., Hamblen, J. W., Hull, T. E., Keenan, T. A., Kehl, W. B., McCluskey, E. J., Navarro, S. O., Rheinboldt, W. C., Schweppe, E. J., Viavant, W., and Young, Jr., D. M. (1968).
Curriculum 68: Recommendations for academic programs in computer science: A report of the acm curriculum committee on computer science.
Commun. ACM, 11(3):151–197.


Bajcsy, R. and Reynolds, C. (2002).
Computer science: The science of and about information and computation.
Commun. ACM, 45(3):94–98.


Denning, P. J. (1985).
What is computer science?
Am. Sci., 73(1):16–19.

References II


Denning, P. J. (2007).
Computing is a natural science.
Commun. ACM, 50(7):13–18.


Forsythe, G. E. (1967).
A university's educational program in computer science.
Commun. ACM, 10(1):3–11.


Hartmanis, J. and Lin, H., editors (1992).
Computing the Future: A Broader Agenda for Computer Science and Engineering.
National Academy Press, Washington, DC, USA.


Newell, A., Perlis, A. J., and Simon, H. A. (1967).
Computer science.
Science, 157(3795):1373–1374.

References III


Newell, A. and Simon, H. A. (1976).
Computer science as empirical inquiry: Symbols and search.
Commun. ACM, 19(3):113–126.


Primiero, G. (2016).
Information in the Philosophy of Computer Science, volume Handobok of the Philosophy of Information of *Routledge Handbooks*.
Routledge.


Wegner, P. (1976).
Research paradigms in computer science.
In *Proceedings of the 2Nd International Conference on Software Engineering*, ICSE '76, pages 322–330, Los Alamitos, CA, USA. IEEE Computer Society Press.