

# Contents

*Declare the things  
that are to come  
hereafter*

ISAIAH 41: 23

Preface	xi
Acknowledgments	xvii
<b>Part I. Preliminaries</b>	<b>1</b>
■ 1. Introduction and Historical Review or, What's It All About?	3
■ 2. Algorithms and Data or, Getting It Done	19
■ 3. Programming Languages and Paradigms or, Getting It Done by Computer	49
<b>Part II. Methods and Analysis</b>	<b>79</b>
■ 4. Algorithmic Methods or, Getting It Done Methodically	81
■ 5. The Correctness of Algorithms or, Getting It Done Right	99



■ 6. The Efficiency of Algorithms or, Getting It Done Cheaply	129
--	-----

### **Part III. Limitations and Robustness 157**

■ 7. Inefficiency and Intractability or, You Can't Always Get It Done Cheaply	159
■ 8. Noncomputability and Undecidability or, Sometimes You Can't Get It Done At All!	191
■ 9. Algorithmic Universality and Its Robustness or, The Simplest Machines That Get It Done	219

### **Part IV. Relaxing the Rules 255**

■ 10. Parallelism, Concurrency, and Alternative Models or, Getting Lots of Stuff Done at Once	257
■ 11. Probabilistic Algorithms or, Getting It Done by Tossing Coins	297
■ 12. Cryptography and Reliable Interaction or, Getting It Done in Secret	317

### **Part V. The Bigger Picture 335**

■ 13. Software Engineering or, Getting It Done When It's Large	337
■ 14. Reactive Systems or, Getting It to Behave Properly Over Time	357
■ 15. Algorithmics and Intelligence or, Are They Better at It Than Us?	379



Postscript	401
Selected Solutions	403
Bibliographic Notes	433
Index	495

This book tells a story. The story concerns the concepts, ideas, methods, and results fundamental to computer science. It is not specifically about computer technology, nor is it about computer programming, though obviously it is heavily influenced by both.

The book is intended to fill a rather disturbing gap in the literature related to the computer revolution. Scores of excellent books can be found on computers themselves, with details of their structure, workings, and operation. There are also numerous books about the art of writing programs for computers in any of a growing number of languages. These books come at a wide range of levels, some aimed at people with no computer-related background at all, and some aimed at the most computer-literate professionals. In addition, there are many books on subjects peripheral to the technology, such as the social and legal aspects of the revolution, as well as books describing the relevance of computers to a variety of application areas. All this comes as no surprise. People are curious about computers, and want to learn how to put them to use. They are typically interested in specific kinds of computers, and often for specific purposes, too.

Then there are textbooks. Indeed, computer science is a fast-growing academic discipline, with ever-larger numbers of potential students knocking at the doors of admission offices. Well-established academic disciplines have a habit of yielding excellent textbooks, and computer science is no exception. Over the years many comprehensive and clearly written textbooks have appeared, containing detailed technical accounts of the subjects deemed appropriate to students of computer science. However, despite the dizzying speed with which some of the *technological* innovations become obsolete and are replaced by new ones, the fundamentals of the science of computation, and hence many of the basic concepts that are considered important in a computer science curriculum, change slowly, if at all. Of course, new technologies and new languages require revisions in scientific emphasis, which are eventually reflected in the scientific literature. However, by and large, there is almost