

STUDENT



EDITION

# ELEMENTARY PASCAL

...of healthful pro-  
perties of  
**Fry's**  
Pure  
Concentrated  
**Cocoa**  
It enables you to  
live the world of  
**DECEMBER.**  
No. 168 XXII.

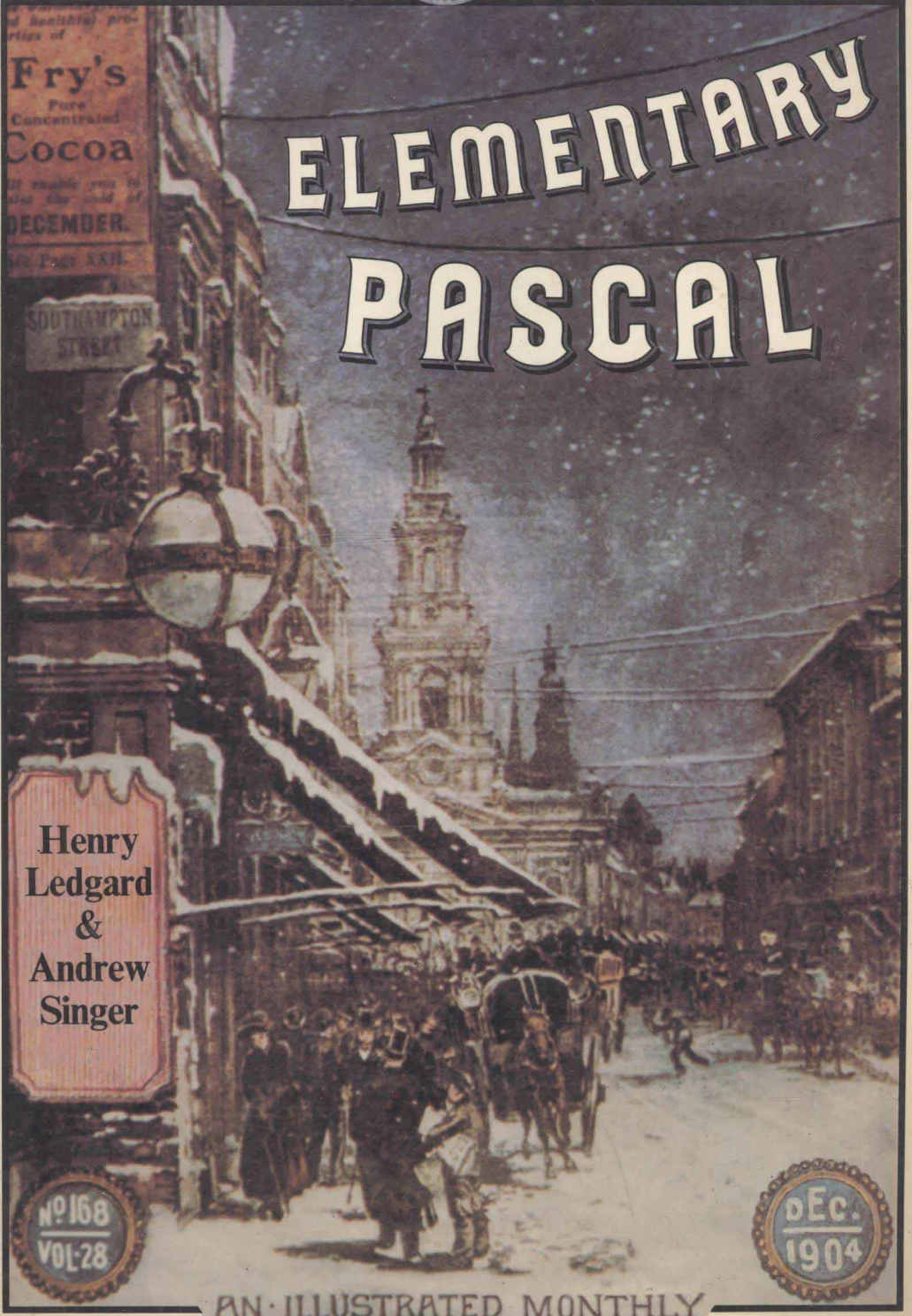
SOUTHAMPTON  
STREET

Henry  
Ledgard  
&  
Andrew  
Singer

NO 168  
VOL 28

DEC.  
1904

AN ILLUSTRATED MONTHLY



# ELEMENTARY PASCAL

As Chronicled by John H. Watson

Edited with Commentaries by

**HENRY LEDGARD**

and

**ANDREW SINGER**



SCIENCE RESEARCH ASSOCIATES, INC.  
Chicago, Palo Alto, Toronto, Henley-on-Thames, Sydney

A Subsidiary of IBM

Acquisition Editor Alan W. Lowe  
Project Editor Ronald Q. Lewton  
Composition and Production Human Factors Limited  
Cover Production Michael Rogondino & Associates

The cover is adapted from an actual issue of the *Strand* magazine for the date shown.

Additional acknowledgments appear in the "Postscript" at the end of the book.

#### **Library of Congress Cataloging in Publication Data**

Ledgard, Henry F., 1943-  
Elementary Pascal.

Includes index.

1. Pascal (computer program language)

I. Singer, Andrew, 1943- II. Title.

QA76.73.P2L394 001.64 '24 81-23183

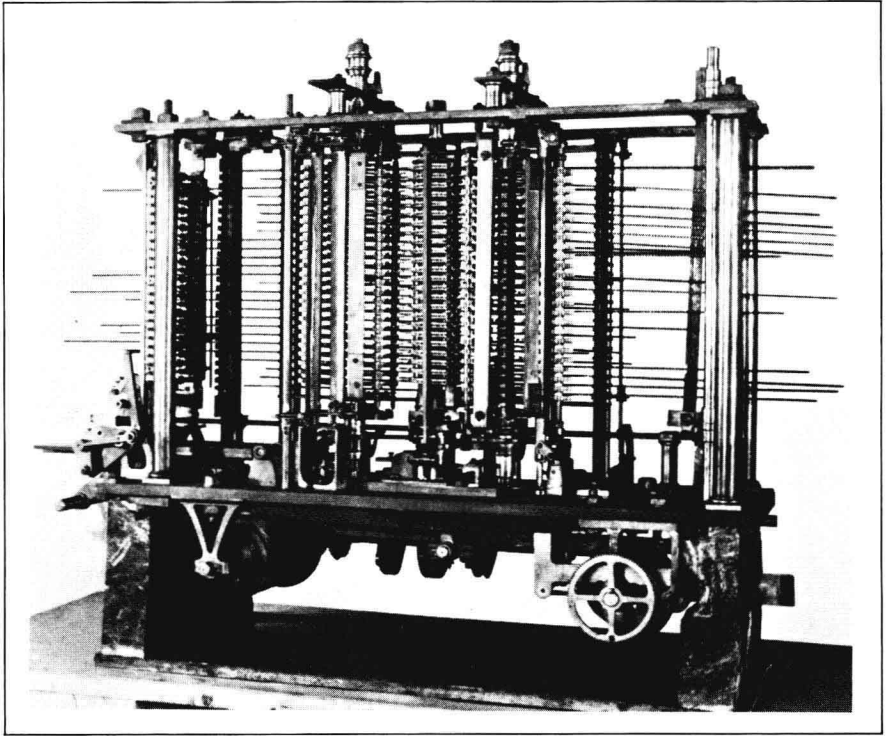
ISBN 0-574-21380-5 AACR2

Copyright © Henry Ledgard, 1982  
All rights reserved

Printed in the United States of America

2 3 4 5 6 7 8 9

# ELEMENTARY PASCAL



THE ANALYTICAL ENGINE

# PREFACE

Putting Sherlock Holmes aside for the moment, this is a very serious text on computer programming in Pascal. Its goal is to provide a thoughtful introduction to programming for the uninitiated.

The orientation of this work is problem solving, not syntax. The reader will notice an absence of syntax equations in the text and an absence of highly detailed discussions of language constructs. But the reader will also notice great care in trying to present the deeper reasons for programming, the essence of concepts, and the need to do useful things. We are committed to the idea that programming languages are subservient to programming, and programming subservient to solving problems.

This work attempts to set a very high standard. The programs, for instance, have been written with extraordinary attention to readability. Program maintenance is one of the greatest problems in software and we emphasize the importance of program quality. The language concepts, for instance, are also motivated with care. We, as authors, have spent considerable time trying to understand some of the deeper concepts in programming, so as to give the reader many of the root ideas.

This work is also a product of our research in human factors. We are of the disposition that much, much greater things can be done in computers to meet the human user half way. The notation, examples, wording, and style of this book reflect these concerns. We have attempted to avoid the jargon of the computer technology wherever possible.

This work is also intuitive. Programming is a complex and demanding skill, and understanding it can be difficult. As such we have emphasized the intuitive aspects, trusting that this will make learning considerably easier. We also trust that, as a text, the intuitive approach makes this book easy to read. To dismiss this work as "too easy" is, in a sense, an indication of the success of this approach.

There are many exercises in this book; devising them was a major work in itself. Our goal was to develop a range of exercises, from easy ones that are matters of program reading or giving short answers to harder ones requiring the writing of a meaningful program. Some are on text-editing, some introduce the computer, some require a five- or ten-

line program, some are fun, and some are full of tricks. At the end of the book, in Chapter XIV, there are some term project exercises.

But why bring in Sherlock Holmes? Because we wanted to present programming in an appealing way as possible. We believe that engaging the learner's interest is fundamental to effective teaching. At the root is our concern for the student as a human being.

The computer that Holmes and Watson use is the Analytical Engine of Charles Babbage. Babbage's Engine, designed just before the era of Sherlock Holmes, was truly a precursor of the modern computer. Moreover, the dialogues between Holmes and Watson are *not* simply idle front matter. It is here, in these episodes, that cornerstones of the technical material is exposed. This is the place where the root ideas are presented, in simple human conversation, unconfused by technical details.

Finally, Holmes and Watson set a human temperament and a focus on the human user of computers. There surely are important.

Generally, each chapter of this book begins with one of the case studies of Sherlock Holmes. Here we see Holmes and Watson solving some problem. Following each case study, the commentary discusses in detail the issues arising from the problem presented by Holmes.

The first three chapters present Holmes's introduction to problem solving and programming. Like Watson, the reader is not expected to be able to duplicate the ideas presented, or even understand them fully. That will follow in due course. Hopefully, though, the general effect of all that follows will emerge. The point of these chapters is to show the essence of programming, reading before writing if you will.

In Chapter IV, Holmes presents the first steps needed to write programs. This is where most traditional books on programming begin. By the end of Chapter VII, the reader should have completed the central issues in writing any computer program. At this point the reader will be well on the way to writing real programs.

The next five chapters should considerably enlarge the basic skills. In these cases, Holmes is dealing with somewhat larger problems and the programming tools needed to solve them effectively.

In the last chapter, Holmes and Watson confront a most difficult case from a computing standpoint. Holmes's solution brings into play almost all the ideas presented in this book.

In writing this text we have not found it necessary to cover all of the Pascal language, although several features omitted in the text are discussed in the problems at the end of each chapter. The portion of Pascal covered in the text is summarized in the Appendix, "Pascal at a Glance." In any case, our version of Pascal closely follows that presented

by Niklaus Wirth, its designer. Be careful — your local dialect may be ever-so-slightly different.

So come, dear reader, the game is afoot.

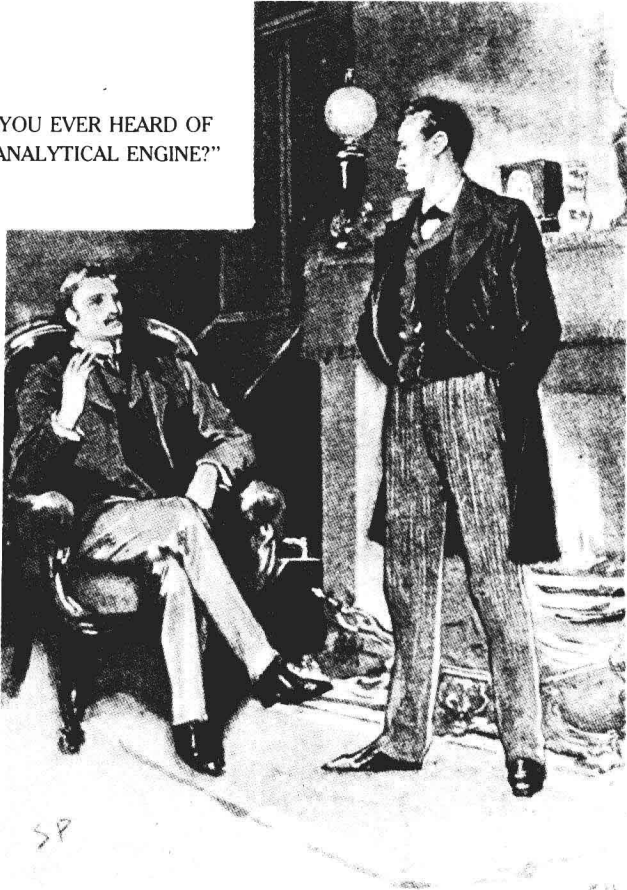
*HENRY LEDGARD*  
*ANDREW SINGER*



# CIRCUMSTANTIAL EVIDENCE



"HAVE YOU EVER HEARD OF  
THE ANALYTICAL ENGINE?"



# CONTENTS

PREFACE .....	xi
TABLE OF ILLUSTRATIVE PROGRAMS .....	xiv
<b>CIRCUMSTANTIAL EVIDENCE</b>	
Chapter I THE ANALYTICAL ENGINE .....	1
1.1 Commentary .....	4
Chapter II MURDER AT THE METROPOLITAN CLUB .....	7
2.1 Algorithms Reviewed .....	17
Chapter III HOLMES GIVES A DEMONSTRATION .....	25
3.1 General Program Structure .....	30
3.2 The Units of a Pascal Program .....	32
3.3 Spacing and Punctuation .....	36
<b>THE GAME AFOOT</b>	
Chapter IV THE ADVENTURE OF THE BATHING MACHINE .....	47
4.1 Variables and Assignment .....	56
4.2 Declaring Variables .....	57
4.3 Declaring Constants .....	59
4.4 Expressions .....	60
4.5 Reading and Writing Information .....	65
Chapter V A STUDY IN CIGAR ASH .....	77
5.1 Compound Statements .....	84
5.2 If Statements .....	85
5.3 Conditions .....	88
5.4 Case Statements .....	91
Chapter VI THE ADVENTURE OF CLERGYMAN PETER .....	99
6.1 Looping .....	109
6.2 While and Repeat Loops .....	109
6.3 For Loops .....	112

Chapter VII	HOLMES'S METHOD REVEALED .....	127
	7.1 An Exception Disproves the Rule .....	127
	7.2 If Matters Become Difficult .....	128
	7.3 A Curious Language of His Own Invention .....	130
	7.4 Programs as Human Communication .....	131
	7.5 The Seemingly Insignificant Blank Space .....	134

### THREE-PIPE PROBLEMS

Chapter VIII	AN ADVERTISEMENT IN THE <i>TIMES</i> .....	145
	8.1 Types and Enumerated Types .....	155
	8.2 Boolean Types .....	160
	8.3 Character Types .....	161
	8.4 Subranges of Types .....	162
	8.5 String Types .....	164
Chapter IX	THE CIPHERED MESSAGE .....	173
	9.1 Arrays .....	182
	9.2 Strings as Arrays .....	186
Chapter X	A STUDY IN CHEMISTRY .....	197
	10.1 Packaging and Subprograms .....	209
	10.2 Parameters .....	211
	10.3 Functions versus Procedures .....	214
	10.4 Global Information .....	217
	10.5 Side Effects .....	218
Chapter XI	THE CORONER'S REPORT .....	233
	11.1 Input and Output .....	245
	11.2 Reading and Writing Files .....	251
	11.3 Some Details .....	253
Chapter XII	THE ADVENTURE OF THE GOLD CHIP .....	263
	12.1 Record Structures .....	273
	12.2 Storing Records .....	274

### THE LAST BOW

Chapter XIII	HOLMES DELIVERS A LECTURE .....	283
	13.1 Revisiting the Top-Down Approach .....	289
Chapter XIV	THE FINAL PROGRAMME .....	295
	14.1 The Remaining Subprograms .....	325

APPENDIX: Pascal at a Glance .....	337
POSTSCRIPT .....	343
INDEX .....	347

## TABLE OF ILLUSTRATIVE PROGRAMS

Algorithmic Clue Search (general features).....	28
Tide Calculation (variables and assignment).....	55
Counting Change (simple I-O, arithmetic expressions).....	68
Cigar Identification (if statements).....	82
Train Times (while and for statements) .....	108
Counting Change Revisited (using loops) .....	114
Date Calculation (enumeration types) .....	154
Message Coding (arrays).....	181
Counting Change Yet Again (arrays and types).....	187
Molecular Weights (functions and procedures).....	205
Report Generation (detailed input-output) .....	242
Data Base Search (records and files).....	269

## CHAPTER I

---

# *The Analytical Engine*

**I**N an incoherent and, as I deeply feel, an entirely inadequate fashion, I have endeavoured to give some account of the remarkable career of Mr. Sherlock Holmes as a criminal investigator and consulting detective. As the reader is undoubtedly well aware, my companion's interests were as broad as Nature herself and he often spoke on an amazing variety of subjects as though he had made a special study of each. In my modest chronicles of the cases that I have had the privilege to share with Sherlock Holmes, I have often alluded to his numerous publications, but I have said nothing before of his unparalleled contributions to the development of the Analytical Engine.

My first introduction to the Analytical Engine was in the late spring, shortly after the conclusion of one of the most ghastly adventures we had ever shared, which I have chronicled under the heading of "The Adventure of the Speckled Band." The entire day Holmes was in a mood that some would call taciturn. He was most unsettled, smoked incessantly, played snatches on his violin, sank into reveries, and hardly answered the casual questions that I put to him. We sat through a silent dinner together, after which, rising from the table, he revealed to me the problem with which he was preoccupied.

"You can never foretell what one mind will come up with, Watson, but you can say with precision what an average person will do. Individuals vary, but percentages remain constant; and while we have not yet grasped the results that the human mind alone can attain, it has its distinct limitations. There are only particular individuals on whom we can rely to produce the same chain of logical argument from one occasion to the next."

"I certainly wouldn't argue with you, Holmes," I replied. "But as yet we haven't found a suitable replacement for human reasoning."

"Oh, on the contrary, Watson," he answered nonchalantly. "Have you ever heard of the Analytical Engine?"

"I know of no substitute for the mind of man."

Holmes chuckled. "Then you must learn of it. It is an ingenious mechanism, a machine that has displayed a considerable talent for deductive reasoning, far superior to the average logician. You recall my intervention in the matter of that notebook floating in the River Cam last month?"

"I am not likely soon to forget the sight of that bloated face staring up at me, Holmes," I replied grimly, considering the sorry state of mankind that such events should come to pass. "What connection has the late professor with this Engine?"

"Well, as you may remember, my investigation led me to the Cavendish laboratories; and it was there that I had occasion to study the Engine, if only briefly. Since then I have been in correspondence with mathematicians at Cambridge who have been conducting experiments with it. Watson, I do not exaggerate when I say that the Analytical Engine is capable of solving, within minutes, complex numerical problems that would keep five of London's finest mathematicians working for hours. Furthermore, it is adept at logic and has a perfect memory for detail.

"The Engine also has its limits," he continued. "It can only undertake problems whose solutions are spelled out in minute detail and that are presented in its own peculiar language."

"Really, Holmes, sometimes you go too far with my patience!" I exclaimed. "You expect me to believe that this device is capable of solving problems, has a perfect memory, and actually speaks a language of its own?"

"No, no, my dear Watson, you take me too literally. The Analytical Engine does indeed have a language of its own, but communications must be written out."

"Now you tell me it can read?"

"In a sense, yes."

I threw up my arms in a desperate gesture and began to rise from the chair.

"I fear I am going too fast for you, Watson. Bear with me for a moment and I shall do my utmost to explain all this to you. Everything I say is true, but let me assure you that the Analytical Engine hardly resembles a human being.



“Its ‘language’ is actually a highly logical code, designed by mathematicians in order to operate the Engine. This code is not difficult to master, but it does require considerable discipline. It has a very small vocabulary, which is nothing to compare with the English tongue. This vocabulary is arranged into statements according to a limited set of rules.

“The major problem in communicating with the Engine is that one must use the utmost care and precision in giving it instructions, for it has no imagination whatsoever and cannot correct even trivial errors in spelling or punctuation. It is, after all, like other machines in that it has no awareness of the tasks that it performs; therefore it will obey the most unreasonable of instructions. For example, if it is told to print the number zero *ad infinitum*, it will continue to do so for hours on end, until a human being finally causes it to stop.”

“But Holmes, how does one give instructions to this Engine?” I asked, scarcely crediting my companion’s remarks thus far and wondering whether perhaps his penchant for cocaine had finally betrayed his reason.

“By writing a set of instructions in code and supplying them mechanically to the Engine. Such a set of instructions is called a *programme*, because it is an orderly and precise procedure for solving a problem. The art of writing programmes is called, reasonably enough, *programming*.”

“Of what relevance is this strange machine to you, Holmes?”

“I intend to employ the Engine whenever possible in my future criminal cases,” he replied. “As you know, I have been rather overburdened with work in recent months, so the Engine’s speed and potential accuracy are most attractive to me. It has a great capacity for dealing with large amounts of information as well.”

“But, Holmes,” I interrupted, “do you truly expect this device, if it is as unimaginative as you say, actually to solve crimes?”

“Not at all, my dear Watson,” said Holmes with a laugh. “I daresay it is not clever enough to replace my brain; but it will be useful for storing information, as well as for performing certain repetitive tasks that absorb too much of my time. Of most interest to me is that it will provide a means of expressing my logical methods in a rigorous form, and perhaps be useful in communicating to others my modest attempts at formulating a Science of Deduction.”

---