# A Machine Made this Book

*ten sketches of computer science*

How do we decide where to put ink on a page to draw letters and pictures? How can computers represent all the world's languages and writing systems? What exactly is a computer program, what and how does it calculate, and how can we build one? Can we compress information to make it easier to store and quicker to transmit? How do newspapers print photographs with grey tones using just black ink and white paper? How are paragraphs laid out automatically on a page and split across multiple pages?

In *A Machine Made this Book*, using examples from the publishing industry, John Whitington introduces the fascinating discipline of Computer Science to the uninitiated.

JOHN WHITINGTON founded a company which builds software for electronic document processing. He studied, and taught, Computer Science at Queens' College, Cambridge. He has written textbooks before, but this is his first attempt at something for the popular audience.

# A Machine Made this Book

*ten sketches of computer science*

John Whitington

COHERENT PRESS

COHERENT PRESS
Cambridge

Published in the United Kingdom by Coherent Press, Cambridge

© Coherent Press 2016

This publication is in copyright. Subject to statutory exception no reproduction of any part may take place without the written permission of Coherent Press.

First published March 2016

*A catalogue record for this book is available from the British Library*

ISBN 978-0-9576711-2-6 Paperback

*by the same author*

PDF Explained (O'Reilly, 2012)
OCaml from the Very Beginning (Coherent, 2013)
More OCaml: Algorithms, Methods & Diversions (Coherent, 2014)

# Contents


Preface v

1 Putting Marks on Paper 1

2 Letter Forms 15

3 Storing Words 27

4 Looking and Finding 41

5 Typing it In 53

6 Saving Space 65

7 Doing Sums 81

8 Grey Areas 97

9 Our Typeface 123

10 Words to Paragraphs 135

Solutions 147

Further Reading 169

Templates 173

Colophon 181

Index 183

# Preface

It can be tremendously difficult for an outsider to understand why computer scientists are interested in Computer Science. It is easy to see the sense of wonder of the astrophysicist, or of the evolutionary biologist or zoologist. We don't know too much about the mathematician, but we are in awe anyway. But Computer Science? Well, we suppose it must have to do with computers, at least. "Computer science is no more about computers than astronomy is about telescopes", the great Dutch computer scientist Edsger Dijkstra (1930–2002), wrote. That is to say, the computer is our tool for exploring this subject and for building things in its world, but it is not the world itself.

This book makes no attempt at completeness whatever. It is, as the subtitle suggests, a set of little sketches of the use of computer science to address the problems of book production. By looking from different angles at interesting challenges and pretty solutions, we hope to gain some insight into the essence of the thing.

I hope that, by the end, you will have some understanding of why these things interest computer scientists and, perhaps, you will find that some of them interest you.

**Chapter 1** starts from nothing. We have a plain white page on which to place marks in ink to make letters and pictures. How do we decide where to put the ink? How can we draw a convincing straight line? Using a microscope, we will look at the effect of putting these marks on real paper using different printing techniques. We see how the problem and its solutions change if we are drawing on the computer screen instead of printing on paper. Having drawn lines, we build filled shapes.

**Chapter 2** shows how to draw letters from a realistic typeface – letters which are made from curves and not just straight lines. We will see how typeface designers create such beautiful shapes, and how we might draw them on the page. A little geometry is involved, but nothing which can't be done with a pen and paper and a ruler. We fill these shapes to draw letters on the page, and deal with some surprising complications.

**Chapter 3** describes how computers and communication equipment deal with human language, rather than just the numbers which are their native tongue. We see how the world's languages may be encoded in a standard form, and how we can tell the computer to display our text in different ways.

**Chapter 4** introduces some actual computer programming, in the context of a method for conducting a search through an existing text to find pertinent words, as we might when constructing an index. We write a real program to search for a word in a given text, and look at ways to measure and improve its performance. We see how these techniques are used by the search engines we use every day.

**Chapter 5** explores how to get a bookful of information into the computer to begin with. After a historical interlude concerning typewriters and similar devices from the nineteenth and early twentieth centuries, we consider modern methods. Then we look at how the Asian languages can be typed, even those which have hundreds of thousands or millions of symbols.

**Chapter 6** deals with compression – that is, making words and images take up less space, without losing essential detail. However fast and capacious computers have become, it is still necessary to keep things as small as possible. As a practical example, we consider the method of compression used when sending faxes.

**Chapter 7** introduces more programming, of a slightly different kind. We begin by seeing how computer programs calculate simple sums, following the familiar schoolboy rules. We then build more complicated things involving the processing of lists of items. By then end of the chapter, we have written a substantive, real, program.

**Chapter 8** addresses the problem of reproducing colour or grey tone images using just black ink on white paper. How can we do this convincingly and automatically? We look at historical solutions to this problem from medieval times onwards, and try out some different modern methods for ourselves, comparing the results.

**Chapter 9** looks again at typefaces. We investigate the principal typeface used in this book, Palatino, and some of its intricacies. We begin to see how letters are laid out next to each other to form a line of words on the page.

**Chapter 10** shows how to lay out a page by describing how lines of letters are combined into paragraphs to build up a block of text. We learn how to split words with hyphens at the end of lines without ugliness, and we look at how this sort of layout was done before computers.

# Acknowledgments

The word list on 67 is from *The Reading Teacher's Book of Lists*, Fourth Edition (Fry, Kress & Fountoukidis), Prentice Hall, 2000. The literary quotations used as example texts in Chapters 3 and the problems in Chapter 6 are from John Le Carré's *Tinker, Tailor, Soldier, Spy*. The text in Chapter 6 is from Franz Kafka's *The Trial*. Likewise in Chapter 10 from his *Metamorphosis*. The hyphenation examples in Chapter 10 are from *Micro-typographic extensions to the TEX typesetting system*, the PhD Thesis of Hàn Thế Thành, Faculty of Informatics, Masaryk University, Brno, October 2000. The cover image shows a Paige Compositor, courtesy of the United States Patent Office. The drawing of French Curves on page 17 was modified from one created by Joshua Certain. The tables on pages 35–39 appear by kind permission of the Unicode Consortium. Unicode is a registered trademark of Unicode, Inc. in the United States and other countries. The facsimile patents on pages 54,55,56,57, and 59 were provided by the United States Patent and Trademark Office. The picture of a Univac keyboard on page 60 appears courtesy of the Retrocomputing Society of Rhode Island. The picture of an IBM Model M keyboard on page 60 was taken by Sal Cangeloso. The woodblock print on page 101 is *Der Formschneider* (The Blockcutter) from the *Panoplia omnium illiberalium mechanicarum* (Book of Trades); it was printed in 1568 and is in the British Museum. The picture on page 102 is a detail of the engraving *Der Kreuzbrunnen zu Marienbad* published by Franz Sartori in 1819. The engraving *Melencolia I* by Albrecht Dürer on page 103 is held at the Minneapolis Institute of Art. The image of a mezzotint plate on page 104 was taken by David Ladmore. The picture of the mezzotint print by Franz Kruger on page 105 is courtesy of the Image Permanence Institute. The Rembrandt etching *The Hundred Guilder Print* on page 106 is held at the Rijksmuseum in Amsterdam. The photograph of film grain on page 108 was provided by Keith Cooper. The electron microscope image of film grain on the same page is courtesy of the University of Rochester. The halftoned picture of the Steinway Hall on East 14th Street in Manhattan on page 109 is from *The Daily Graphic*, December 2nd, 1873. The examples of Zapfino alternate glyphs on page 127 are based on the instructions of Dario Taraborelli. The tables of Palatino Linotype on pages 131–133 were produced using the eponymous typeface from Monotype GmbH. The sketches of metal typesetting on pages 142–143 are from an unknown early 20th century book. The photograph on page 144 was taken by Tom Garnett at the Print Shop of the Cambridge Museum of Technology, Cambridge, UK. All other photographs and images were created by the author. Scrabble is a trademark of Hasbro, Inc.

# A Machine Made this Book

*ten sketches of computer science*