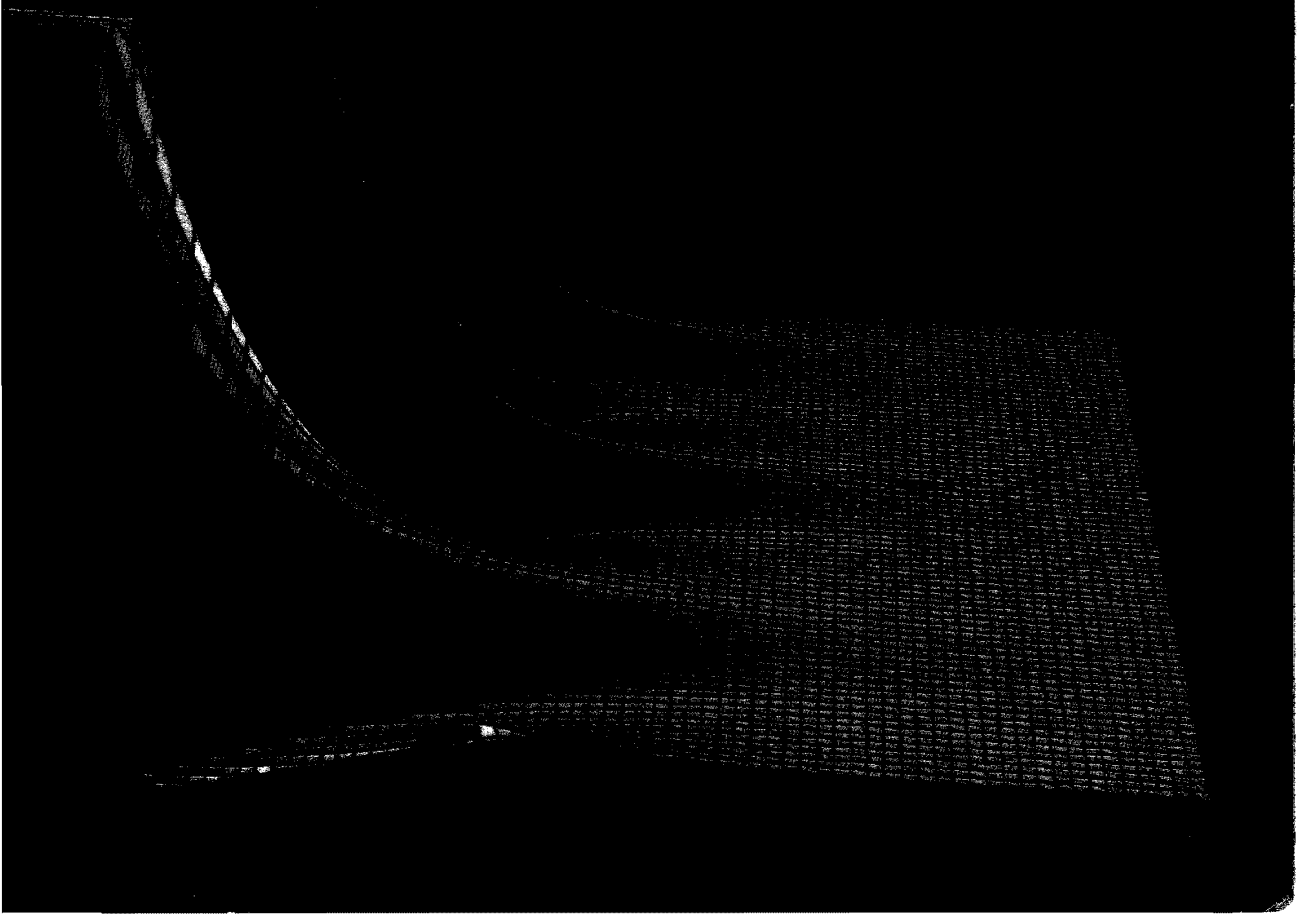
Addison-Wesley Publishing Company

Mathematica™

A System for Doing Mathematics by Computer

Stephen Wolfram



Mathematica™

A System for Doing Mathematics by Computer

Stephen Wolfram

Mathematica was designed and implemented by:

Stephen Wolfram, Daniel Grayson, Roman Maeder, Henry Cejtin, Theodore Gray, Stephen Omohundro, David Ballman and Jerry Keiper

Addison-Wesley Publishing Company, Inc.

The Advanced Book Program

Redwood City, California • Menlo Park, California
Reading, Massachusetts • New York • Amsterdam
Don Mills, Ontario • Sydney • Bonn • Madrid
Singapore • Tokyo • San Juan • Wokingham, United Kingdom

Library of Congress Cataloging-in-Publication Data

Wolfram, Stephen
Mathematica: A System for Doing Mathematics / Stephen Wolfram
p. cm.
Includes index.
ISBN 0-201-19334-5; 0-201-19330-2 (pbk).
1. *Mathematica* (Computer program) 2. Mathematics–Data processing.
I. Title.
QA76.95.W65 1988
510'.28'553—dc19 87-24628
CIP

ABCDEFGHIJ-DO-898

About the Developers of *Mathematica*

Stephen Wolfram was responsible for the overall design of *Mathematica*. He also wrote much of the basic code in the kernel of *Mathematica*, as well as the text of this book.

Wolfram was born in London in 1959. He was educated at Eton, Oxford and Caltech, and received his PhD in theoretical physics from Caltech in 1979.

After two years on the faculty at Caltech, Wolfram moved to the Institute for Advanced Study in Princeton. In 1986, he moved from there to the University of Illinois, to become Director of the Center for Complex Systems Research, and Professor of Physics, Mathematics and Computer Science.

Wolfram's research has covered many areas of physics, mathematics and computer science. His early work (1976–1980) was primarily in high-energy physics, quantum field theory and cosmology.

In recent years, Wolfram has been a leader in the development of the new field of complex systems research. Complex systems research is concerned with the study of systems whose component parts are simple, but whose overall behavior is complex. Beginning in 1982, Wolfram pioneered the application of a class of computational models known as cellular automata to complex systems that arise in mathematics and physics. Wolfram's work on cellular automata has been widely applied in many different fields. It has led to new models for biological and physical pattern formation, and to new approaches to studies of chaos and randomness. In 1984, Wolfram invented a fast encryption system based on cellular automata, and in 1985, he was co-inventor of a new approach to computational fluid dynamics. In 1986, he founded the journal *Complex Systems*.

Much of Wolfram's work has involved the use and development of new computer techniques. In 1980–1981, Wolfram led the development of the SMP computer algebra system. More recently, Wolfram has worked on the development of algorithms for massively parallel computers.

Wolfram has been a consultant for many organizations, including Los Alamos National Lab, Bell Labs and Thinking Machines Corporation. He received a MacArthur Prize Fellowship in 1981.

Daniel R. Grayson wrote many of the mathematical parts of *Mathematica*, including arbitrary-precision arithmetic, equation solving, matrix manipulations, power series and elliptic functions. He also wrote the precompiler for the extension of the C programming language used in developing *Mathematica*.

Grayson is currently Professor of Mathematics at the University of Illinois. After receiving his PhD in mathematics from MIT in 1976, Grayson worked at Columbia University and the Institute for Advanced Study, before moving to the University of Illinois in 1982. Grayson's main research interest is algebraic K-theory, a branch of mathematics which brings together ideas from algebraic geometry, linear algebra and number theory. Grayson has used computers extensively to study conjectures in number theory. Before working on *Mathematica*, Grayson developed an interactive computer system for number theory research.

Roman E. Maeder was responsible for symbolic integration, polynomial factorization, and other polynomial operations in *Mathematica*. Maeder received his PhD from ETH in Zurich in 1986, with a thesis on the mathematical theory of programming languages. Since 1983, Maeder has worked on computer algebra, and its applications to mathematics education. He has organized "mathematical laboratories" for graduate courses in computer mathematics, first at ETH and more recently in the Mathematics Department at the University of Illinois.

Henry Cejtin wrote the final versions of many central routines in *Mathematica*, and helped to rationalize many aspects of the overall design of the system. Cejtin's work has alternated between pure mathematics and software development. His main mathematical research has been in algebraic geometry. He received a PhD in mathematics from Northwestern University in 1985, and has

taught at Northwestern and the University of Illinois in Chicago. Cejtin has also been involved in a number of major software development projects. In 1983, he was responsible for parts of the UNIX-like operating system developed by Mark Williams Company.

Theodore Gray created the front ends for *Mathematica* on the Macintosh and other computers. Before working on *Mathematica*, Gray did graduate work in theoretical chemistry at Berkeley. In 1985, he was the author of a Macintosh system for teaching linear algebra.

Stephen M. Omohundro wrote the three-dimensional graphics code for *Mathematica*. Omohundro received his PhD in mathematical physics from Berkeley in 1985. In 1985–86 he worked at Thinking Machines Corporation on algorithms for massively parallel computation. He was co-designer of the extension of LISP used on the Connection Machine computer. In 1986, Omohundro moved to the University of Illinois to become Assistant Professor in the Department of Computer Science, and a member of the Center for Complex Systems Research. His current research is primarily concerned with the development of general algorithms for machine learning. He is currently working on his second book, tentatively entitled "Geometric Learning in Vision, Graphics and Robotics".

David Ballman was responsible for many aspects of the external system interface for *Mathematica*. Ballman has been involved in a range of computer hardware and software projects, first at the University of Minnesota, and, more recently, at the University of Illinois.

Jerry Keiper wrote the code for evaluation of special functions (`Gamma`, `Zeta`, `BesselJ`, etc.) in *Mathematica*, as well as for various numerical operations (`NIntegrate`, `NSum`, `FindRoot`, etc.). Before working on *Mathematica*, Keiper earned two master's degrees in mathematics, did research on the Riemann zeta function, and built pipe organs.

About the Cover Illustration

The front cover of this book shows a three-dimensional plot of the Riemann zeta function $\zeta(z)$ in the complex plane. The height of the surface is given by the absolute value of $\zeta(z)$; the color is related to the phase of $\zeta(z)$.

The Riemann zeta function arises particularly in number theory, where it gives an analytical representation of certain aspects of the distribution of prime numbers. One of the most commonly studied features of the zeta function is the Riemann hypothesis, which states that all complex zeroes of the zeta function must lie on the "critical line" $\mathrm{Re}(z) = \frac{1}{2}$.

The picture on the cover is obtained by looking up the critical line. The complex zeroes of the zeta function appear as dips nestled in the side of the "mountain" on the left.

The basic form of the *Mathematica* input used to generate the plot was `Plot3D[Abs[Zeta[x + I y]], {x, -2, 6}, {y, 2, 35}]`. Various options to specify resolution, coloring, viewing angle and so on, were added.

The plot took about three minutes to produce on a Sun 3/260 computer. The POSTSCRIPT version of the plot was converted into a bit map, and then color separated for printing by Spectral Effects Ltd. using a Silicon Graphics Iris computer.