Undergraduate Texts in Mathematics

E. J. LeCuyer

Introduction to College Mathematics with A Programming Language

Springer-Verlag
New York Heidelberg Berlin

Edward J. LeCuyer
Western New England College
Department of Mathematics
Springfield, MA 01119
USA

AMS Subject Classifications: 00-01, 00A05, 26-01, 26-04, 26A06, 26A09

Library of Congress Cataloging in Publication Data

LeCuyer, E J
Introduction to college mathematics with A Programming Language

(Undergraduate texts in mathematics)
Includes indexes.
1. Mathematics—1961- 2. Mathematics—Data processing.
3. APL (computer program language) 1. Title.
QA39.2.L4 510 78-2054

9 8 7 6 5 4 3 2 1

ISBN-13:978-1-4613-9424-2 e-ISBN-13:978-1-4613-9422-8
DOI: 10.1007/978-1-4613-9422-8

To my wife, Carol, and children, Karen,
Michael, and Todd, whose love, patience,
and encouragement made this work possible.
To Howard Peelle for all of his help.

Preface

The topics covered in this text are those usually covered in a full year's course in finite mathematics or mathematics for liberal arts students. They correspond very closely to the topics I have taught at Western New England College to freshmen business and liberal arts students. They include set theory, logic, matrices and determinants, functions and graphing, basic differential and integral calculus, probability and statistics, and trigonometry. Because this is an introductory text, none of these topics is dealt with in great depth. The idea is to introduce the student to some of the basic concepts in mathematics along with some of their applications. I believe that this text is self-contained and can be used successfully by any college student who has completed at least two years of high school mathematics including one year of algebra. In addition, no previous knowledge of any programming language is necessary.

The distinguishing feature of this text is that the student is given the opportunity to learn the mathematical concepts via **A P**rogramming **L**anguage (APL). APL was developed by Kenneth E. Iverson while he was at Harvard University and was presented in a book by Dr. Iverson entitled *A Programming Language*[1] in 1962. He invented APL for educational purposes. That is, APL was designed to be a consistent, unambiguous, and powerful notation for communicating mathematical ideas. In 1966, APL became available on a time-sharing system at IBM. Today, APL is gaining wide acceptance in such fields as business, insurance, scientific research, and education. The reason for this is that APL is one of the most concise, versatile, and powerful computer programming languages yet developed. Programs requiring several steps in other computer languages become very

[1] *A Programming Language* by Kenneth E. Iverson, New York: John Wiley and Sons, (1962).

concise in APL, if a program is needed at all. This is both because many primitive functions are available directly on the APL keyboard and because such APL operations as + and × can be applied to arrays of any size (as well as to scalars). Yet, in spite of power and sophistication of APL, it is not a difficult language to learn. One can use APL to solve mathematical problems immediately after only a few minutes of instruction.

Conventional mathematical notation and APL notation are presented *in parallel* throughout the text. Thus, if one desires, it is possible to ignore the APL and still use this text as a standard survey-of-mathematics text. Alternatively, one may use the text in conjunction with an APL terminal. APL notation corresponds closely to standard mathematical notation, and many mathematical processes are executed very easily in APL. By using the computer, the student can save a great deal of time doing tedious calculations and can concentrate more on the principles and concepts of the mathematics. In addition, the APL programs tend to reinforce these principles and concepts. It is my experience that by using APL, the student may learn the mathematical concepts better while finding the learning of mathematics meaningful and enjoyable. As an important bonus, he will be learning a powerful programming language which he will then be able to use in many other courses as well as in the "real world."

The mathematical concepts and the APL notation are presented in parallel throughout the text because I believe that the APL can best be learned as needed in the development of the mathematics rather than as a separate topic. However, it might also be quite useful to have an APL reference for those who have not previously been exposed to the APL language. Therefore, I have included as an appendix an introduction to APL, including the writing and revising of APL programs. This appendix can be quickly perused at the start of the course and then referred to as needed throughout the course.

Finally, I would like to express my appreciation to Dr. Howard A. Peelle of the University of Massachusetts for his encouragement and his numerous valuable suggestions on ways to improve upon this text. Also, I would like to thank the many students at the University of Massachusetts and at Western New England College who used the preliminary versions of this test for their preserverance, encouragement, and suggestions.

July, 1977 Edward J. LeCuyer, Jr.

Contents