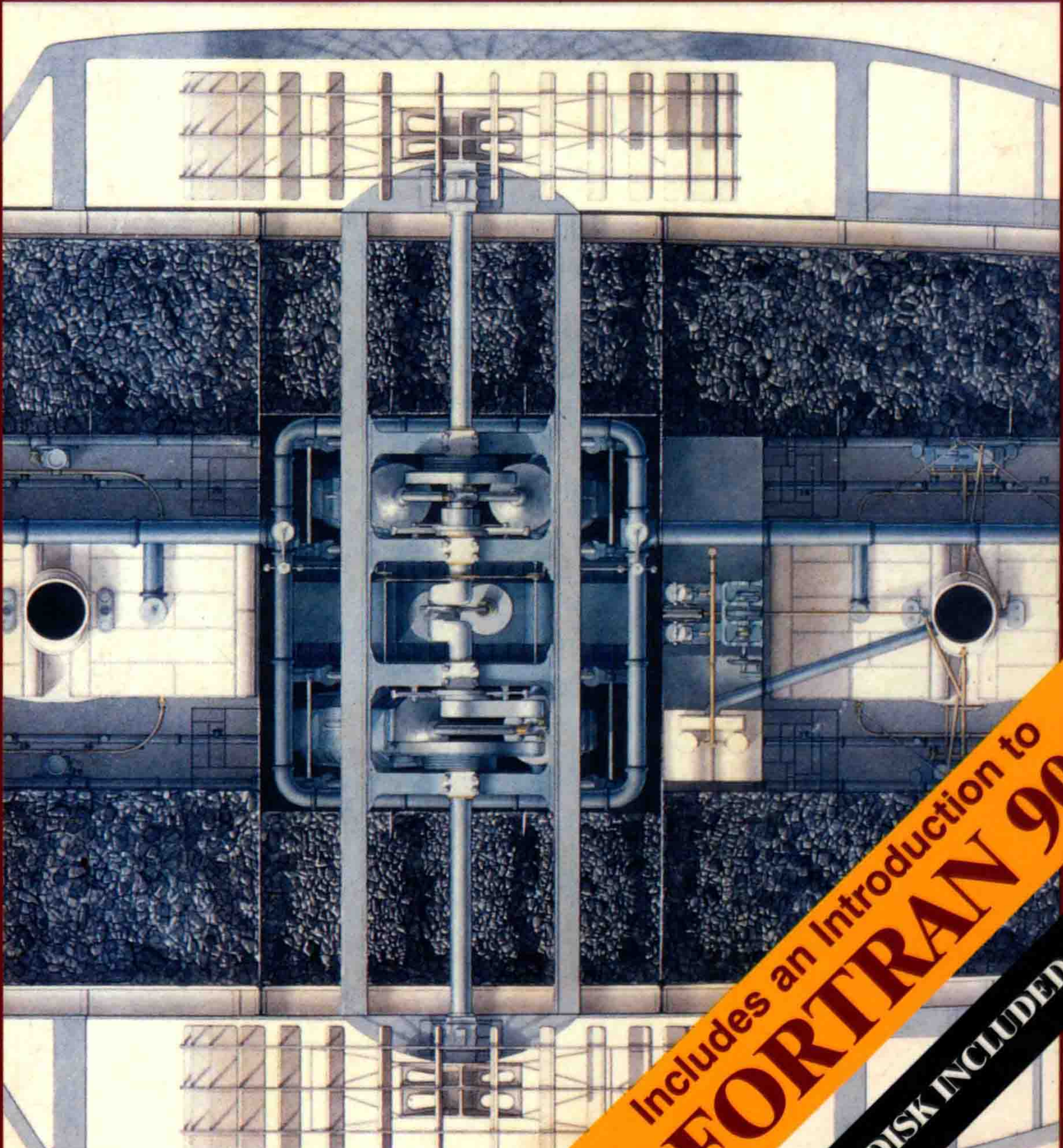
LARRY NYHOFF • SANFORD LEESTMA

FORTRAN 77

and Numerical Methods for Engineers and Scientists



Includes an Introduction to FORTRAN 90

DISK INCLUDED

FORTRAN 77 and Numerical Methods for Engineers and Scientists

Larry Nyhoff

Calvin College

Sanford Leestma

Calvin College

Prentice Hall, Englewood Cliffs, New Jersey 07632

Library of Congress Cataloging-in-Publication Data

Nyhoff, Larry R.
FORTRAN 77 and numerical methods for engineers and scientists /
Larry Nyhoff, Sanford Leestma.
p. cm.
Includes index.
ISBN 0-02-388741-9
1. FORTRAN 77 (Computer program language) I. Leestma, Sanford.
II. Title.
QA76.73.F25N88 1995 94-7462
005.13'3—dc20 CIP

Publisher: Alan Apt
Project Manager: Kimberly Kim
Cover Designer: Marianne Frasco
Cover Photo: Science Museum/Science and Society Picture Library

© 1995 by Prentice-Hall, Inc.
A Simon & Schuster Company
Englewood Cliffs, New Jersey 07632

The author and publisher of this book have used their best efforts in preparing this book. These efforts include the development, research, and testing of the theories and programs to determine their effectiveness. The author and publisher shall not be liable in any event for incidental or consequential damages in connection with, or arising out of, the furnishing, performance, or use of these programs.

All rights reserved. No part of this book may be reproduced, in any form
or by any means, without permission in writing from the publisher.

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

0-02-388741-9

Prentice Hall International (UK) Limited, *London*
Prentice Hall of Australia Pty. Limited, *Sydney*
Prentice Hall Canada Inc., *Toronto*
Prentice Hall Hispanoamericana, S.A., *Mexico*
Prentice Hall of India Private Limited, *New Delhi*
Prentice Hall of Japan, Inc., *Tokyo*
Simon & Schuster Asia Pte., Ltd., *Singapore*
Editora Prentice Hall do Brasil, Ltda., *Rio de Janeiro*

Dedicated to the memory of

DAVID JOHNSTONE

who is missed by all of us who knew and admired this sincere, kind, and gentle man. We thank God for him and for the opportunity of working with him and for his friendship.

PREFACE

FORTRAN, now 40 years old, is a language that is used throughout the world to write programs for solving problems in science and engineering. Since its creation in the late 1950s, it has undergone a number of modifications that have made it a very powerful yet easy-to-use language. These modifications, however, led to a proliferation of different dialects of FORTRAN, which hindered program portability. Since some uniformity was desirable, the American National Standards Institute (ANSI) published the first FORTRAN standard in 1966. In the years following, extensions to this standard version of FORTRAN were developed, some of which came into common use. It became apparent that many of these features should be incorporated into a new standard. This updated ANSI FORTRAN standard (ANSI X3.9-1978), popularly known as FORTRAN 77, is the basis for this text. A new standard has recently been finalized, and the version of FORTRAN—FORTRAN 90—specified by this standard has many new features. At the end of each chapter and in Chapter 14 we describe some of these additions to FORTRAN.

FORTRAN is one of the most widely used programming languages for solving problems in science and engineering. This text describes these applications in the examples and exercises. It contains more than 60 complete examples and over 400 exercises, both written and programming exercises, chosen from areas that are relevant to science and engineering students. Copies of all programs and subprograms marked in the text with a disk icon are on the data disk that accompanies this text.

A special feature of this text is its emphasis on numerical methods. These methods are highlighted in 15 special NUMERICAL METHODS sections, which we have marked with a special icon. These sections contain descriptions of and programs for a variety of methods that are particularly useful in solving science and engineering problems. These include the following:

- Least squares approximations.
- Root finding: Bisection method, Newton's method, false position, secant method, fixed-point iteration.
- Linear systems: Matrices, Gaussian elimination, Jacobi iteration, Gauss-Seidel iteration.
- Derivatives and differences: First and second differences.
- Numerical integration: Rectangle method, trapezoidal method, Simpson's rule, Romberg integration, Gaussian quadrature.

此为试读,需要完整PDF请访问: www.ertongbook.com

- Differential equations: Euler's method, Runge-Kutta method, multistep methods, predictor-corrector methods.
- Interpolation: Linear, quadratic, cubic, Lagrange, and piecewise.
- Function approximation: Method of undetermined coefficients, cubic splines, Taylor polynomials.

Although this book gives a complete presentation of FORTRAN 77, it is more than just a programming manual. The main reason for learning a programming language is to use the computer to solve problems. Consequently, the basic steps in problem solving and program development are discussed and illustrated: (1) problem analysis and specification, (2) algorithm development, (3) program coding, (4) program execution and testing, and (5) program maintenance. We also feel that an intelligent user of the computer must have some elementary understanding of the manner in which a computer operates, how it stores information, how it processes data, and what some of its capabilities and limitations are. For this reason the text also contains a brief sketch of the history of computers and a simple description of a computer system, including techniques for representing data and machine instructions in binary form.

The text also emphasizes the importance of good structure and style in programs. In addition to describing these concepts in general, it contains a large number of complete examples. Each of these consists of a description of the algorithm using pseudocode and/or flowcharts together with a program and sample run. These are intended to demonstrate good algorithm design and programming style. At the end of each chapter a Programming Pointers section summarizes the main points regarding structure and style as well as language features presented and potential problems that beginning programmers may experience.

Supplementary Materials

There are two supplementary materials available to instructors who adopt this text:

- A solutions manual.
- Disks containing solutions to programming exercises.

Acknowledgments

We express our sincere appreciation to all who helped in any way in the preparation of this text, especially our editors, Betsy Jones at Macmillan and Alan Apt at Prentice Hall. We also thank Ferenc Szidarovszky for contributing engineering exercises that are solved using numerical methods. And, of course, we must once again thank our wives, Shar and Marge, whose love and understanding have kept us going through another year of textbook writing, and to our kids and grandkids, Jeff and Dawn, Jim, Julie, Joan, Rebecca, Michelle and Paul, Sandy and Lori, and Michael, for not complaining about the times that their needs and wants were slighted because of our busyness. Above all, we give thanks to God for giving us the opportunity, ability, and stamina to prepare this text.

L. N.
S. L.

CONTENTS

1 INTRODUCTION AND HISTORY 1

1.1 The History of Computing 2

The Mechanization of Arithmetic 2
The Stored Program Concept 6
Mechanical Computers 6
Early Electronic Computers 8
Modern Computers 11
Computer Software 13
Summary 14

1.2 Computer Organization 14

Computing Systems 14
Memory Organization 15
Number Systems 16
Data Storage 18
Instruction Processing 22

Exercises 24

2 PROGRAM DEVELOPMENT 29

2.1 Problem Analysis and Specification 31

Problem 1: Radioactive Decay 31
Problem 2: Pollution Index 32
Problem 3: Mean Time to Failure 33

2.2 Design 33

Problem 1: Radioactive Decay—Sequential Structure 34
Problem 2: Pollution Index—Selection Structure 37
Problem 3: Mean Time to Failure—Repetition Structure 38

2.3 Coding 41

Variables 41
Types 42
Operations 42

Assignment 42
Input/Output 43
Comments 43
Program Composition 43

2.4 Verification and Validation 45

2.5 Software Engineering 47

Problem Analysis and Specification 48
Design 48
Program Coding 49
Verification and Validation 52
Maintenance 52

Exercises 53

3 BASIC FORTRAN 59

3.1 Data + Algorithms = Programs 60

3.2 Program Format 61

3.3 Constants and Variables 62

Constants 62
Identifiers 64
Variables 65
Named Constants: The **PARAMETER** Statement 67
Variable Initialization: The **DATA** Statement 69

Exercises 70

3.4 Arithmetic Operations and Functions 72

Operations 72
Functions 76

Exercises 77

3.5 The Assignment Statement 78

Exercises 84

3.6 Input/Output 86

List-Directed Output 88
List-Directed Input 89

3.7 Program Composition 93

3.8 Example: Acidity of a Diluted Mixture 95

Exercises 99

3.9 NUMERICAL METHODS: Arithmetic Errors 104

Overflow/Underflow Errors 105
Roundoff Errors 106
Truncation Errors 108

Programming Pointers 109

Program Design 109
Potential Problems 110
Program Style 113

FORTRAN 90 Features 113

Program Format 114
Constants and Variables 114
Arithmetic Operations and Functions 115
Program Composition 115

4 STRUCTURED PROGRAMMING 117

4.1 Sequential Structure: Statement Sequences 118

4.2 The LOGICAL Data Type 119

Simple Logical Expressions 120
The Effect of Roundoff Errors 122
Compound Logical Expressions 123

4.3 Example: Logical Circuits 126

Exercises 128

4.4 The IF Selection Structure 131

Simple **IF** Statement 131
Pollution Index Problem 134
Example: Quadratic Equations—Version 1 136
Nested **IF** Constructs 139
Example: Quadratic Equations—Version 2 139
IF–ELSE IF Construct 142
Example: Quadratic Equations—Version 3 143

Exercises 146

4.5 Repetition Structure: DO Loops 149

Example: A Table of Points on a Curve 152
Example: A Multiplication Table 153

4.6 The While Repetition Structure 155

Example: Summation 155
Sentinel-Controlled While Loops 158

4.7 The DO WHILE Statement 159

Example: Summation 160
Example: Calculating the Mean Time to Failure 162

4.8 Implementing While Loops in Standard FORTRAN 164

Example: Summation 164
Example: Calculating the Mean Time to Failure 166

4.9 A Posttest Repetition Structure 168

Example: Calculating the Mean Time to Failure 169

4.10 Program Testing and Debugging Techniques 172

An Example 172
Trace Tables 174
Debugging 175
Modifying and Testing a Program 176
Summary 178

Exercises 179

4.11 NUMERICAL METHODS: An Overview 183

4.12 NUMERICAL METHODS: Least Squares Line; Solving Equations 184

Curve Fitting: Least Squares Line 184
Solving Equations: The Divide-and-Average Algorithm 188

Exercises 190

Programming Pointers 192

Program Design 192
Potential Problems 193
Program Style 196

FORTRAN 90 Features 197

5 INTRODUCTION TO SUBPROGRAMS AND MODULAR PROGRAMMING 203

5.1 Functions 204

Library Functions 204
Function Subprograms 205
Example: Voltage Across a Capacitor 207
Argument Association 209
Example: Function of Several Variables 209
Example: Pollution Index 210
Local Identifiers—The Factorial Function 211
Example: Poisson Probability Function 213
The **SAVE** Statement 215
Example: Simulation Using a Random Number Generator 215
Statement Functions 222

Exercises 223

5.2 NUMERICAL METHODS: Root Finding 228

The Bisection Method 228
Newton's Method 231

Exercises 236

5.3 NUMERICAL METHODS: Integration 239

The Rectangle Method 240
The Trapezoidal Method 243
Simpson's Rule 243

Exercises 244

5.4 NUMERICAL METHODS: Differential Equations 246

Euler's Method 246
Runge–Kutta Method 250

Exercises 250

5.5 Subroutine Subprograms 254

Example: Displaying an Angle in Degrees 255
Example: Displaying an Angle in Degrees–Minutes–Seconds Format 258
Example of a Subroutine That Returns Values: Converting Coordinates 262
Argument Association 264

Exercises 267

5.6 Examples of Modular Programming and Top-Down Design: Beam Deflection and Checking Academic Standing 269

Example 1: Beam Deflection 269
Example 2: Checking Academic Standing 277

5.7 Functions and Subroutines as Arguments 296

The **EXTERNAL** Statement 296
The **INSTRINSIC** Statement 298

5.8 The COMMON Statement 300

Blank Common 301
Named Common 304

Exercises 305

Programming Pointers 307

Program Design 307
Potential Problems 308
Program Style 309

FORTRAN 90 Features 310

6 INPUT/OUTPUT 315

6.1 Formatted Output 316

Control Characters 318
Integer Output—The **I** Descriptor 319
Real Output—The **F** Descriptor 321
Real Output—The **E** Descriptor 322
Double Precision Output—The **D** Descriptor 323
Logical Output—The **L** Descriptor 324
Character Output 324
Positional Descriptors—**X** and **T** 325
Repeating Groups of Format Descriptors 326
The Slash (/) Descriptor 327
Scanning the Format 328

此为试读，需要完整PDF请访问：www.ertongbook.com

6.2 Example: Printing Tables of Computed Values 330

Exercises 331

6.3 Formatted Input 333

Integer Input 334
Real Input 335
Character Input 337
Logical Input 338
Skipping Columns of Input 338
Multiple Input Lines 339

6.4 The WRITE Statement and the General READ Statement 340

The **WRITE** Statement 340
The General **READ** Statement 343

6.5 Introduction to File Processing 345

Opening Files 346
Closing Files 347
File Input/Output 347
The **IOSTAT** = Clause 348
The **END** = Clause 349
File-Positioning Statements 350

6.6 Examples: Time, Temperature, Pressure, and Volume Readings; Run-Time Formatting 351

Example 1: Time, Temperature, Pressure, and Volume Readings 351
Example 2: Run-Time Formatting 353

6.7 Miscellaneous Input/Output Topics 357

The **G** Descriptor 357
Scale Factors 357
The **BN** and **BZ** Descriptors 359
The **S, SP** and **SS** Descriptors 360
The **H** Descriptor 360
The **TL** and **TR** Descriptors 360
List-Directed Input 361

Exercises 362

Programming Pointers 365

FORTRAN 90 Features 366

7 ONE-DIMENSIONAL ARRAYS 369

7.1 Introduction to Arrays; Subscripted Variables 370

Problem 370
Arrays 373
Array Declarations 376
Address Translation 378

7.2 Input/Output of Arrays 379

Input/Output Using a **DO** Loop 379
Input/Output Using the Array Name 383
Input/Output Using Implied **DO** Loops 385

7.3 Example: Processing a List of Failure Times 387

7.4 Array Processing 389

Assigning Values to Arrays 389
Arrays as Arguments 391
Arrays in Common 394

7.5 Examples: Frequency Distributions and Bar Graphs; Averages of Grouped Data 395

Example 1: Frequency Distributions and Bar Graphs 395
Example 2: Averages of Grouped Data 400
Exercises 403

7.6 Vector Processing 410

Vector Representation 410
Vector Operations 411
A Vector Library 415
Exercises 415

7.7 Sorting and Searching 416

Sorting 416
Searching 421
Exercises 428

7.8 NUMERICAL METHODS: Interpolation 430

Linear Interpolation 430
Quadratic Interpolation 430
Cubic Interpolation 431
Lagrange Interpolation in General 433
Piecewise Interpolation 434
Exercises 435

Programming Pointers 436

FORTRAN 90 Features 439

8 MULTIDIMENSIONAL ARRAYS 443

8.1 Introduction to Multidimensional Arrays and Multiply Subscripted Variables 444

8.2 Processing Multidimensional Arrays 447

Input/Output Using **DO** Loops 449
Input/Output Using the Array Name 453
Input/Output Using Implied **DO** Loops 454
Multidimensional Arrays as Arguments 456
Multidimensional Arrays in Common 460

8.3 **Examples: Pollution Table, Oceanographic Data Analysis 460**

Example 1: Pollution Table 460
Example 2: Oceanographic Data Analysis 464

Exercises 469

8.4 **NUMERICAL METHODS: Linear Systems and Matrices 474**

Linear Systems 474
Matrices 486
A Matrix Library 496

Exercises 496

Programming Pointers 504

FORTRAN 90 Features 505

9 THE COMPLEX DATA TYPE 509

9.1 **The Complex Data Type 510**

Representation of Complex Numbers 510
The **COMPLEX** Type Statement 511
Operations on Complex Numbers 511
Complex Functions 514
Complex I/O 515

9.2 **NUMERICAL METHODS: Solutions of Equations 516**

Polynomial Equations 516
Linear Systems 521

Exercises 521

Programming Pointers 523

FORTRAN 90 Features 524

10 MORE NUMERICAL METHODS 525

10.1 **NUMERICAL METHODS: Root Finding 526**

The Method of False Position 526
The Secant Method 529
Fixed-Point Iteration 531

Exercises 533

10.2 **NUMERICAL METHODS: Integration 534**

Romberg Integration 534
Gaussian Quadrature 537

Exercises 540

10.3 **NUMERICAL METHODS: Differential Equations 541**

Multistep Methods 541
Predictor–Corrector Methods 542

Higher-Order Differential Equations 542

Exercises 544

10.4 NUMERICAL METHODS: Derivatives and Differences 545

First Differences 545
Second Differences 548

Exercises 549

10.5 NUMERICAL METHODS: Linear Systems 551

Jacobi Iteration 552
Gauss–Seidel Iteration 556

Exercises 557

10.6 NUMERICAL METHODS: Function Approximation 558

Method of Undetermined Coefficients 558
Cubic Splines 559
Taylor Polynomials 561

Exercises 562

11 ADVANCED CHARACTER DATA 565

11.1 Character Data and Operations 566

Character Operations 566
Output 569
Input 570

11.2 The INDEX and LEN Functions 571

INDEX 571
LEN 572
Example: Text Editing 573

11.3 Character Comparison 578

The **LLT, LLE, LGT,** and **LGE** Functions 579
The **ICHAR** and **CHAR** Functions 580
Example: Cryptography 580

11.4 Examples: Plotting Graphs, Density Plots, and Contour Maps 582

Example 1: Plotting Graphs 582
Example 2: Density Plots and Contour Maps 586

11.5 Examples: Lexical Analysis, Reverse Polish Notation 593

Example 1: Lexical Analysis 593
Example 2: Reverse Polish Notation 599

Exercises 613

Programming Pointers 621

FORTRAN 90 Features 622

12 FILE PROCESSING 625

12.1 **The OPEN, CLOSE, and INQUIRE Statements 626**

Opening Files 626
Closing Files 630
The **INQUIRE** Statement 631

12.2 **File Input/Output and Positioning 631**

File Input 631
Example 1: Direct-Access Inventory File 634
File-Positioning Statements 636
File Output 636
Example 2: Merging Files 637
Example 3: External Sorting: Mergesort 641
Unformatted Files 642
Internal Files 643

12.3 **Example: Inventory Update 645**

Exercises 657

FORTRAN 90 Features 659

13 ADDITIONAL FORTRAN FEATURES 661

13.1 **The STOP and PAUSE Statements 662**

13.2 **The IMPLICIT Statement 662**

13.3 **Other Control Statements: Arithmetic IF, Computed GO TO, Assigned GO TO 664**

The Arithmetic **IF** Statement 664
The Computed **GO TO** Statement 665
The Assigned **GO TO** Statement 666

13.4 **More About COMMON and Block Data Subprograms 666**

Other **COMMON** Features 666
Block Data Subprograms 669

13.5 **The EQUIVALENCE Statement 670**

13.6 **Alternate Entries and Returns 674**

The **ENTRY** Statement 674
Alternate Returns 676

FORTRAN 90 Features 677

14 NEW DIRECTIONS IN FORTRAN 90 679

14.1 **Modules 680**

14.2 **Derived Data Types 683**

14.3 **Pointers and Linked Structures 688**

APPENDIXES

A ASCII and EBCDIC 699

B Sample Files 705

An Inventory File 705
An Inventory-Update File 706
A Student File 708
A Student-Update File 710
A User's File 712
A User-Update File 714
A Least Squares File 715

C Program Composition 717

D Generic and Specific Names of Functions 719

E Answers to Selected Exercises 723

INDEX OF PROGRAMMING EXERCISES 741

INDEX OF EXAMPLES AND SAMPLE PROGRAMS 749

INDEX 753

此为试读，需要完整PDF请访问：www.ertongbook.com