

C.1851

BIBLIOTHEQUE DU CERIST

**PROBLEM
SOLVING
AND STRUCTURED
PROGRAMMING
IN
PASCAL**



ELLIOT B. KOFFMAN



Reference Guide to Pascal Statements

Statement	Example of Use
<i>program heading</i>	program GUIDE (INPUT, OUTPUT, INFILE, OUTFILE);
<i>comment</i>	{This section shows examples of Pascal statements.} (* Comments are ignored by Pascal. *)
<i>constant declaration</i>	const
<i>integer</i>	STRINGSIZE = 20;
<i>character</i>	BLANK = ' ';
<i>string</i>	SCHOOL = 'TEMPLE UNIVERSITY';
<i>real</i>	DEANSLIST = 3.5; PROBATION = 1.0;
<i>type declaration</i>	type
<i>enumerated</i>	COLLEGE = (BUSINESS, ARTS, EDUCATION, GENERAL);
<i>subrange</i>	STUDENTRANGE = 1..100;
<i>string</i>	STRING = packed array [1..STRINGSIZE] of CHAR;
<i>pointer</i>	CLASSPOINTER = ^STUDENT;
<i>record</i>	STUDENT = record NAME : STRING; GPA : REAL; INCOLLEGE : COLLEGE; NEXTSTU : CLASSPOINTER end; {STUDENT}
<i>array</i>	MAJORARRAY = array [STUDENTRANGE] of COLLEGE;
<i>file</i>	STUFILE = file of STUDENT;
<i>set</i>	GRADESET = set of 'A'..'Z';
<i>variable declaration</i>	var
<i>record</i>	CURSTU : STUDENT;
<i>set</i>	GRADES : GRADESET;
<i>text file</i>	INFILE : TEXT;
<i>file</i>	OUTFILE : STUFILE;
<i>pointer</i>	CLASSLIST : CLASSPOINTER;
<i>array</i>	MAJOR : MAJORARRAY;
<i>character</i>	NEXTCH : CHAR;
<i>integer</i>	I, COUNTPROBATION : INTEGER;
<i>declaring function</i> <i>with BOOLEAN</i> <i>result</i>	function MEMBER (NEXTCH : CHAR; TESTSET : GRADESET) : BOOLEAN; {Returns TRUE if NEXTCH is a member of TESTSET.}
<i>assignment (BOOLEAN)</i> <i>set membership</i>	begin {MEMBER} MEMBER := NEXTCH in TESTSET {Is NEXTCH in set?} end; {MEMBER}

(continued on last page)



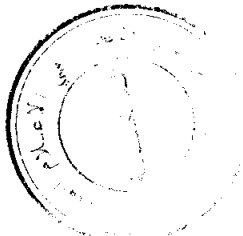
BIBLIOTHEQUE DU CERIST

BIBLIOTHEQUE DU CERIST

BIBLIOTHEQUE DU CERIST

Problem Solving and Structured Programming in Pascal

SECOND EDITION



BIBLIOTHEQUE DU CERIST

C
1851

Problem Solving and Structured Programming in Pascal

SECOND EDITION

ELLIOT B. KOFFMAN

Temple University



ADDISON-WESLEY PUBLISHING COMPANY, INC.

Reading, Massachusetts • Menlo Park, California •
Don Mills, Ontario • Wokingham, England • Amsterdam •
Sydney • Singapore • Tokyo • Mexico City • Bogotá •
Santiago • San Juan

James T. DeWolf, Sponsoring Editor

Hugh Crawford, Manufacturing Supervisor

Robert Forget, Art Editor

Fran Palmer Fulton, Production Editor

Karen M. Guardino, Production Manager

Richard Hannus, Hannus Design Associates, Cover Design

Maureen Langer, Text Designer

Turbo Pascal[®] is a registered trademark of Borland International.

UCSD Pascal[™] is a trademark of the Regents of the University of California.

Library of Congress Cataloging in Publication Data

Koffman, Elliot B.

Problem solving and structured programming in
Pascal.

Includes index.

1. Pascal (Computer program language) 2. Structured
programming. I. Title.

QA76.73.P2K63 1985 001.64'24 84-16811

ISBN 0-201-11736-3

ISBN 0-201-21450-4 School

4572

Reprinted with corrections, June 1985

Copyright © 1985, 1981 by Addison-Wesley Publishing Company, Inc. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher. Printed in the United States of America. Published simultaneously in Canada.

To my family—Caryn, Richard, Deborah, and Robin Koffman,
for their constant love and understanding.

To my parents—Edward and Leah Koffman, for all that they
have given me.

BIBLIOTHEQUE DU CERIST

Preface

There have been many changes in the way the first course in Computer Science is taught since the first edition of this book was published in 1981. During the past two years I have been the chairman of the ACM Task Force that has been studying these changes with an eye towards updating the description of the recommended first course for Computer Science majors (CS1).¹ Parallel with this effort, the Educational Testing Service (ETS) published a set of guidelines for an Advanced Placement course in Computer Science.² The text has been completely revised and reorganized to conform to both of these guidelines.

This text can be used in any introductory programming course that emphasizes a careful, disciplined approach to programming in Pascal. Since the Advanced Placement course is a full year course, this text covers more material than would normally be completed in one semester. The additional material on searching and sorting algorithms (Chapter 9) and dynamic data structures (Chapter 10) are optional advanced topics in CS1 and would normally be deferred until CS2.

As in the first edition, the primary goal of this text is to illustrate and teach problem solving through the stepwise development of algorithms. To facilitate this, procedures are introduced much earlier in this edition. There are also several large case studies throughout the text that integrate

¹Koffman, E., Miller, P., and Wardle, C. Recommended Curriculum for CS1, 1984. Communications ACM 27, 10 (Oct., 1984), 998-1001.

²Advanced Placement Program of the College Board, Advanced Placement Course Description: Computer Science. Educational Testing Service, Princeton, NJ, 1983.

topics and illustrate their application in a substantial programming problem with multiple procedures. Many new examples and programming assignment projects are provided.

Some of the important features of this new edition are:

Early introduction of procedures: Procedures without parameters are introduced in Chapter 2 and are used for self-contained operations that require no global variable access (no side-effects). Procedure parameters are discussed in Chapter 3. Early coverage of procedures will enable students to practice the top-down approach from the beginning and to become more adept at program modularization.

Interactive programming: The emphasis is on modern technology and interactive programming. The majority of examples are written as interactive programs; however, students are shown early how to convert these programs to run in a batch environment. There are some batch-oriented examples as well.

New chapter on recursion: There is a new chapter on recursion that provides many examples of recursive procedures and functions. Additional algorithms for searching and sorting arrays are also provided in this chapter.

Arrays: Single and multidimensional arrays are covered in one chapter instead of in two as in the first edition. Similarly, all material on records is covered in a single chapter.

New expanded case studies: There are a number of new, larger case studies within the text. The solutions to the case studies are all carefully developed. System structure charts are used to represent the flow of control and data between modules in a program system.

Spiral approach: A spiral approach is used to preview topics such as the `if` statement, `for` statement, and input/output. Features are introduced as needed rather than overwhelming a student by providing all the details at once.

Pedagogical aids:

- Self-check Exercises are provided at the end of most sections. Solutions to selected exercises are provided at the end of the text.
- Each chapter ends with a Chapter Review section that includes a summary, a table of new Pascal statements, and review questions.
- *Boxed material:* Syntax display boxes are used to describe the syntax of each new Pascal statement as it is introduced, while Program Style boxes discuss the importance of good programming style.
- *Error warnings:* Each chapter ends with a discussion geared toward helping students prevent and correct common programming errors. Several sections discuss debugging techniques.
- *Program comments:* All programs are carefully commented. Loop invariants and assertions are shown for some loops. For easy identification, the first and last line of each procedure or program is in blue type.

New design: The page layout is larger providing more white space and

the overall tone is more user-friendly. The book has been completely redesigned with an eye towards making it easier for students to find figures, examples, programs, and special display boxes. A second color is used both to improve the appearance of the text and to clarify illustrations.

Pascal dialects: ANSI standard Pascal is covered in the text. Common extensions are described in appendixes on ISO standard Pascal, UCSD Pascal and Turbo Pascal.

Reference appendixes: There are also appendixes covering Pascal language elements, syntax diagrams, character codes, and error messages.

Complete instructor's manual: An Instructor's Manual provides a discussion of how to teach the concepts in each chapter. Sample test questions will be included as well as answers to all exercises, chapter review questions, and the Programming Projects found at the end of each chapter.

Transparency masters: A set of 131 transparency masters illustrating important concepts is available upon request.

Acknowledgments

Many people participated in the development of this text. The principal reviewers were most essential in finding errors and suggesting improvements. They include: William Eccles, University of South Carolina; Frank Friedman, Temple University; David Hannay, Union College; Helen Holzbaur, Temple University; Abraham Kandel, Florida State University; Raymond Kirsch, LaSalle College; David Moffat, North Carolina State University; Tom Nelson, North Carolina State University; Richard Rinewalt, University of Texas at Arlington; Paul Ross, Millersville University of Pennsylvania; Chris Speth, Western Kentucky University; Caroline Wardle, Boston University; Charles C. Weems, Jr., University of Massachusetts at Amherst; and Brad Wilson, Western Kentucky University. I am grateful to all of them for their considerable efforts.

Towards the beginning of this project, several faculty and members of the Addison-Wesley sales and marketing staffs participated in focus groups to discuss the first programming course in Pascal. These discussions were helpful in providing direction to the text and clarifying its organization. The faculty are: Linda Ottenstein, Michigan Tech University; David Neusse, University of Wisconsin at Eau Claire; Richard Rinewalt, University of Texas at Arlington; Ruth Barton, Michigan State University; and Howard Edelman, West Chester State University.

Finally, a number of faculty reviewed and commented on preliminary sections of the text. These faculty include: Gideon Frieder, University of Michigan; Gary Ford, University of Colorado; Abraham Kandel, Florida State University; Paul Hanna, Florida State University; M. Main, University of Colorado; Kent Wooldridge, California State University at Chico; Richard St. Andre, Central Michigan University; C. E. Wolf, Iowa State University; Angela Wu, American University; Yure Gurevich, University of

Michigan; Amir Foroudi, State University of New York at Fredonia; Morris Rang, II, Western Illinois University; Peggy R. Ayres, Linn-Benton Community College; Muhammed H. Chaudhary, Millersville University of Pennsylvania; Stanley Thomas, Wake Forest University; R. J. Del Zoppo, Jamestown Community College; David Rosenlof, Sacramento City College; George Beekman, Oregon State University; George Witter, Western Washington State University; J. M. Adams, New Mexico State University; John Lushbough, University of South Dakota; Dave Valentine, State University of New York at Potsdam; Dennis Martin, State University of New York at Brockport; Chris J. Dovolis, University of Minnesota; Barbara Smith-Thomas, University of North Carolina at Greensboro; Barent Johnson, University of Wisconsin at Oshkosh; Carl Wehner, University of Northern Iowa; Aboalfazl Salimi, Embry-Riddle University; Larry Wilson, Old Dominion University; Cary Laxer, Rose-Hulman Institute of Technology; J. Mailen Kootsey, Duke University; Jerry Waxman, City University of New York at Queens; Bruce J. Klein, Grand Valley State College; Eris Pas, Duke University; Gove Effinger, Bates College; Krishna Moorthy, Framingham State College; Brian Johnson, University of New Hampshire; and John Goda, Georgia Institute of Technology.

There were also many people involved with the actual production of the text. From Addison-Wesley, James DeWolf was the sponsoring editor and recruited reviewers, provided input and suggestions during the writing stage, and coordinated with the production staff. Bill Gruener also was the publisher with overall responsibility for the text. Karen Guardino was the production manager and saw to it that the book was able to meet a very tight production schedule. Maureen Langer refined the design of the text. In Philadelphia, Fran Palmer Fulton served as the Production Editor and coordinated and supervised the typesetting of the manuscript. I am grateful to all of them for their involvement and extra efforts to get this book published on schedule.

*Philadelphia, PA
December 1984*

E.B.K.

Contents

Introduction to Computers and Programming 1

- 1.1 Electronic Computers Then and Now 2
- 1.2 Components of a Computer 3
- 1.3 Problem Solving and Programming 6
- 1.4 Programming Languages 7
- 1.5 Processing a High-level Language Program 9
- 1.6 Introduction to Pascal 11
- 1.7 General Form of a Pascal Program 22
- 1.8 Using the Computer 26
- 1.9 Formatting Program Output 29
- 1.10 Introduction to Data Types 34
- 1.11 Common Programming Errors 36
- 1.12 Chapter Review 39

Problem Solving 45

- 2.1 Representing and Refining Algorithms 46
- 2.2 Using Procedures for Subproblems 52
- 2.3 Decision Steps in Algorithms 59
- 2.4 Tracing a Program or Algorithm 67
- 2.5 Problem Solving Strategies 69

2.6	Repetition in Programs	72
2.7	Generalizing a Solution	79
2.8	Repeating a Program Body	83
2.9	Debugging and Testing Programs	87
2.10	Common Programming Errors	88
2.11	Chapter Review	89

Control Statements 95

3.1	Syntax Diagrams	96
3.2	The if Statement Revisited	97
3.3	The while Statement	104
3.4	Procedure Parameters	111
3.5	Adding Data Flow Information to Structure Charts	128
3.6	Nested Procedures and Scope of Identifiers	132
3.7	Case Studies	136
3.8	Debugging a Program System	150
3.9	Common Programming Errors	151
3.10	Chapter Review	152

Simple Data Types 159

4.1	Constant Declarations	160
4.2	Numeric Data Types—REAL and INTEGER	161
4.3	Functions in Arithmetic Expressions	169
4.4	BOOLEAN Variables, Expressions, and Operators	175
4.5	Character Variables and Functions	180
4.6	Introduction to Programmer-defined Data Types	188
4.7	Input/Output Revisited	193
4.8	Case Study	201
4.9	Common Programming Errors	206
4.10	Chapter Review	207

More Control Statements 217

5.1	The case Statement	218
5.2	Set Values in Decisions	221
5.3	The General for Statement	223

5.4	The repeat Statement	225
5.5	Nested Loops	229
5.6	User-defined Functions	234
5.7	Case Studies	240
5.8	Common Programming Errors	251
5.9	Chapter Review	252

Arrays 263

6.1	Declaring and Referencing Arrays	264
6.2	Arrays with Integer Subscripts	266
6.3	Case Study	270
6.4	Manipulating Entire Arrays	276
6.5	Reading Part of an Array	282
6.6	General Arrays	283
6.7	Character Strings	288
6.8	Multidimensional Arrays	295
6.9	Case Study	302
6.10	Common Programming Errors	313
6.11	Chapter Review	313

Records 325

7.1	Declaring a Record	326
7.2	Manipulating Individual Fields of a Record	328
7.3	Manipulating an Entire Record	331
7.4	Arrays of Records	334
7.5	Case Study	335
7.6	Searching an Array	342
7.7	Sorting an Array	344
7.8	General Data Structures	348
7.9	Record Variants	352
7.10	Manipulating Strings Stored in Records	357
7.11	Common Programming Errors	363
7.12	Chapter Review	364

Sets and Files 371

- 8.1 Set Data Type and Set Operators 372
- 8.2 RESET, REWRITE, and the File Position Pointer 379
- 8.3 TEXT Files 381
- 8.4 Case Studies 385
- 8.5 User-defined File Types 396
- 8.6 Case Study—File Merge 401
- 8.7 Case Study—Data Base Inquiry 405
- 8.8 File Buffer Variable 411
- 8.9 Common Programming Errors 416
- 8.10 Chapter Review 417

Recursion, Searching, and Sorting 425

- 9.1 The Nature of Recursion 426
- 9.2 Recursive Procedures 432
- 9.3 Recursive Functions 439
- 9.4 Binary Search of an Array 449
- 9.5 Searching by Hashing 453
- 9.6 Additional Sorting Algorithms 457
- 9.7 Case Study—The Quicksort Algorithm 462
- 9.8 Common Programming Errors 467
- 9.9 Chapter Review 468

Pointer Variables and Dynamic Data Structures 475

- 10.1 The NEW Statement and Pointer Variables 476
- 10.2 Understanding Dynamic Allocation 480
- 10.3 Introduction to Linked Lists 481
- 10.4 Manipulating Linked Lists Using Pointer Variables 483
- 10.5 Case Study—Maintaining a Linked List 487
- 10.6 Stacks and Queues 500
- 10.7 Multiple-linked Lists and Trees 506
- 10.8 Case Study—Maintaining a Binary Search Tree 511
- 10.9 Common Programming Errors 518
- 10.10 Chapter Review 519

■ **Appendix A: Reserved Words, Standard Identifiers,
Operators, Functions, and
Procedures Ap-1**

■ **Appendix B: Additions and Extensions to
Pascal Ap-5**

■ B.1 Additional Features of ANSI/IEEE Pascal

■ B.2 ISO Pascal

■ B.3 UCSD Pascal

■ B.4 TURBO Pascal

■ **Appendix C: Pascal Syntax Diagrams Ap-19**

■ **Appendix D: Character Sets Ap-29**

■ **Appendix E: Error Number Summary Ap-31**

■ **Answers to Selected Exercises Ans-1**

■ **Index I-1**