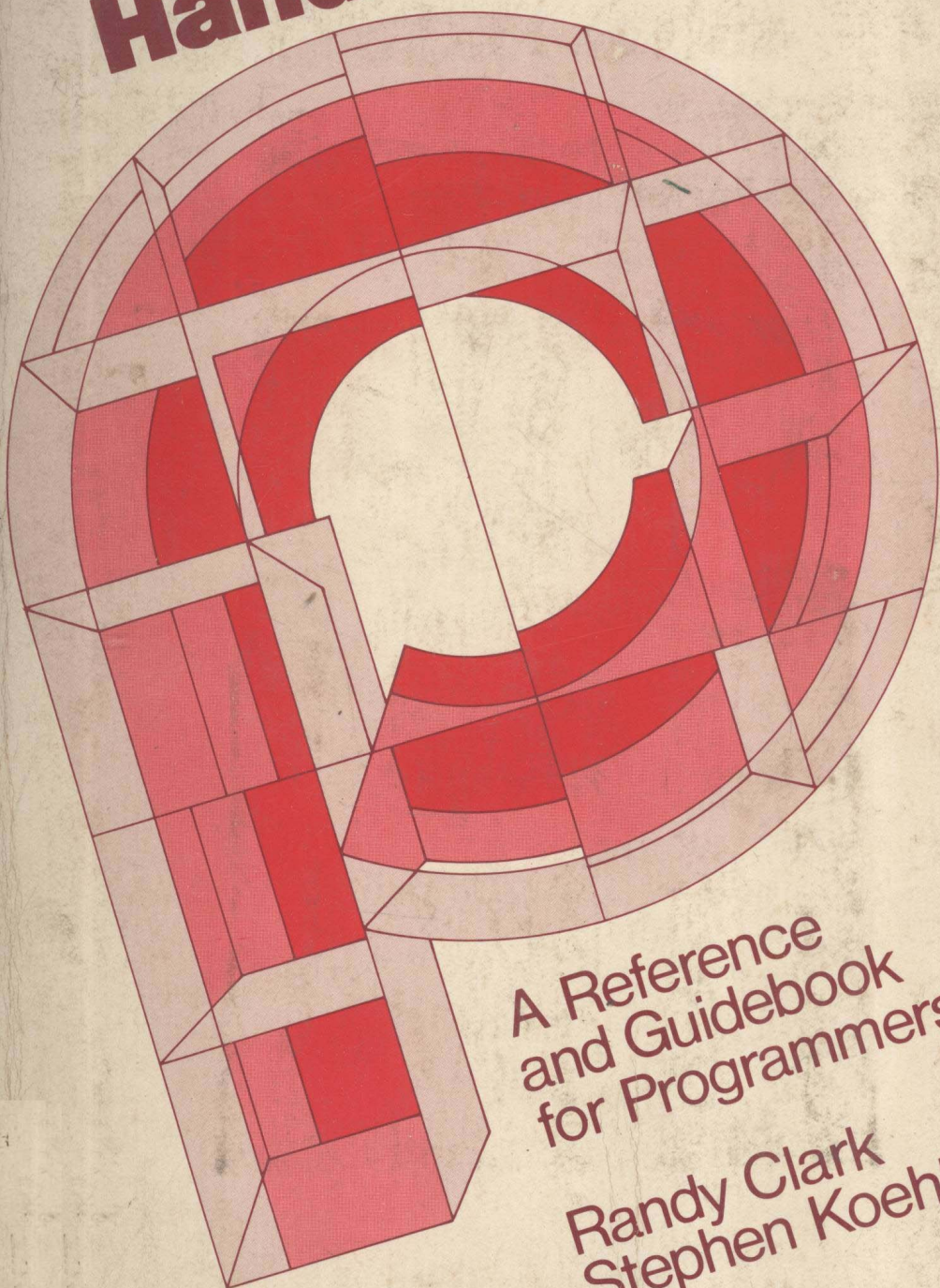


The UCSD Pascal Handbook

A REWARD BOOK



\$15.95



A Reference
and Guidebook
for Programmers

Randy Clark
Stephen Koehler

The UCSD Pascal Handbook

A Reference and Guidebook for Programmers

**RANDY CLARK
STEPHEN KOEHLER**

Clark, Randy.

The UCSD Pascal handbook.

(Prentice-Hall software series)

Bibliography: p.343

Includes index.

1. PASCAL (Computer program language)
2. UCSD Pascal Software System (Computer system) I. Koehler, Stephen. II. Title.

III. Title: U.C.S.D. Pascal handbook.

IV. Series.

QA76.73.P2C56 001.64'24 81-17795

ISBN 0-13-935544-8

AACR2

ISBN 0-13-935536-7 (pbk.)

Editorial/production supervision by *Linda Paskiet*

Cover design by *Jorge Hernandez*

Manufacturing buyer: *Gordon Osbourne*

UCSD PASCAL and UCSD p-System are all trademarks of the regents of the University of California. Use thereof in conjunction with any goods or services is authorized by specific license only and any unauthorized use is contrary to the laws of the State of California.

© 1982 by PRENTICE-HALL, INC.,
Englewood Cliffs, New Jersey 07632

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

ISBN 0-13-935544-8

ISBN 0-13-935536-7 {PBK}

PRENTICE-HALL INTERNATIONAL, INC., *London*
PRENTICE-HALL OF AUSTRALIA PTY. LIMITED, *Sydney*
PRENTICE-HALL OF CANADA, LTD., *Toronto*
PRENTICE-HALL OF INDIA PRIVATE LIMITED, *New Delhi*
PRENTICE-HALL OF JAPAN, INC., *Tokyo*
PRENTICE-HALL OF SOUTHEAST ASIA PTE. LTD., *Singapore*
WHITEHALL BOOKS LIMITED, *Wellington, New Zealand*

The UCSD Pascal Handbook

PRENTICE-HALL SOFTWARE SERIES
BRIAN W. KERNIGHAN, ADVISOR

for Dick and Joan Conn

Preface

Before adding another book to the plethora of books about Pascal, we had to convince ourselves that it was worth the effort, that we were not reinventing the wheel. In our own work, for a long time, both of us have needed a complete and unified description of UCSD Pascal. Until now, there has been none.

This book addresses virtually all users of UCSD Pascal -- from the programmer who would learn Pascal, and the beginner who needs a reference, to the expert.

We have not tried to teach programming. That would require a radically different organization. The motivations to write a UCSD Pascal reference were, to us at least, more immediate.

UCSD Pascal is a language we have enjoyed using. Its embodiment of the concepts of structured programming makes it simple and useful. We believe in well-structured program development, and our comments and advice will make that clear. But we have no interest in being doctrinaire; we don't want programmers to follow dogma; we want them to write correct and viable programs. Neither Pascal nor structured programming is a culmination of the programmer's art; in time we expect to see still more powerful developments. But it will be a very long time before the full potential of Pascal is explored, and the techniques of its use fully mastered.

With those attitudes in mind, we hope that the present volume will be a useful tool for many users, for many years.

We are glad to be able to offer our thanks to a large number of people:

Al Irvine, for instigating the project and providing the environment for its completion;

All those who reviewed the manuscript, or portions of it: Gail and Paul Anderson, Winsor Brown, Rich Gleaves, Rick Grunsky, Sharon Koehler, Tom Marrs, Mark Overgaard, David Reisner, and Dale Sevier;

Niklaus Wirth, for starting it all; Roger Sumner, for his contributions to UCSD Pascal; and the community of Pascal users "without whom all of this would not have been necessary;"

Finally, all of those people who have offered friendship and encouragement. Most will be as pleased to see this book in print as the authors themselves!

DAUGHTER: Daddy, why do things get in a muddle?

FATHER: What do you mean? Things? Muddle?

DAUGHTER: Well, people spend a lot of time tidying things, but they never seem to spend time muddling them. Things just seem to get in a muddle by themselves. And then people have to tidy them up again.

Gregory Bateson
in Steps to an Ecology of Mind

The UCSD Pascal Handbook

Table of Contents

Preface	xiii
Introduction	1
A Thumbnail History of Pascal	1
Pascal at UCSD	2
This Handbook	5
 PART I UCSD Pascal/A Definition	
I. A Synopsis	9
1. The Form of a Program	9
2. Declaration	10
3. Evaluation	11
4. Action	12
5. Communication	15
6. Modularity	15
7. Miscellany	16
8. Execution	17
II. Lexical Conventions	19
0. Conventions of this Handbook	20
0.1 EBNF	21
1. The Character Set	22
2. Symbols	23
2.1 Special Symbols	23
2.2 Reserved Words	24
3. Identifiers	25
3.1 Definition	25
3.2 Uniqueness	25
3.3 Predeclared Identifiers	27
4. Comments	29
5. Token Boundaries	31
6. Indentation and Legibility	32
III. Data Types and Operations	35
1. Introduction	36
1.1 Constants, Variables, and Expressions	36
1.2 Assignment	37
1.3 Evaluating Expressions	38
1.4 The Use of Data Types	39

2.	Simple Data Types	40
2.1	Integer	41
2.1.1	Format	41
2.1.2	Comparisons	41
2.1.3	Operations	42
2.1.4	Routines	44
	<u>abs</u>	44
	<u>sqr</u>	45
2.2	Real	46
2.2.1	Format	46
2.2.2	Comparisons	47
2.2.3	Operations	47
2.2.4	Routines	48
	<u>abs</u>	49
	<u>sqr</u>	49
	<u>sqrt</u>	49
	<u>sin</u>	49
	<u>cos</u>	49
	<u>arctan</u>	49
	<u>exp</u>	50
	<u>ln</u>	50
	<u>log</u>	50
	<u>pwroften</u>	50
2.3	Long Integer	51
2.3.1	Format	51
2.3.2	Comparisons	51
2.3.3	Operations	51
2.3.4	Routines	52
2.4	Boolean	54
2.4.1	Format	54
2.4.2	Comparisons	54
2.4.3	Operations	54
2.4.4	Routines	55
2.5	Characters	56
2.5.1	Format	56
2.5.2	Comparisons	56
2.5.3	Operations	57
2.5.4	Routines	57
2.6	Special-Purpose Types	58

2.7	Type Conversions	59
2.7.1	Compatibility	59
2.7.2	Conversion Routines	59
	<u>trunc</u>	59
	<u>round</u>	60
	<u>odd</u>	60
	<u>ord</u>	60
	<u>chr</u>	60
	<u>str</u>	60
3.	Scalars and Subranges	61
3.1	Scalars	61
3.2	Subranges	62
3.3	Predeclared Scalar Types	63
3.4	Scalar and Subrange Routines	63
	<u>ord</u>	63
	<u>pred</u>	64
	<u>succ</u>	64
4.	Structured Types	65
4.1	Arrays	65
4.1.1	Format	66
4.1.2	Comparisons	67
4.1.3	Operations	67
4.1.4	Routines	68
	<u>fillchar</u>	68
	<u>moveleft</u>	69
	<u>moveright</u>	70
	<u>scan</u>	70
4.2	Strings	71
4.2.1	Format	71
4.2.2	Comparisons	72
4.2.3	Operations	72
4.2.4	Routines	72
	<u>concat</u>	72
	<u>copy</u>	73
	<u>delete</u>	74
	<u>insert</u>	74
	<u>length</u>	75
	<u>pos</u>	75
4.3	Sets	76
4.3.1	Format	76
4.3.2	Comparisons	77
4.3.3	Operations	77
4.3.4	Routines	78

4.4	Records	79
4.4.1	Format	79
4.4.2	Comparisons and Operations	81
4.4.3	Variant Records	81
4.4.4	The With Statement	82
5.	Dynamic Types	84
5.1	Files	84
5.1.1	Format	84
5.1.2	Standard File Handling	86
	<u>eof</u>	87
	<u>eoln</u>	87
	<u>put</u>	87
	<u>get</u>	88
	<u>reset</u>	88
	<u>rewrite</u>	89
5.2	Pointers	90
5.2.1	Format	90
5.2.2	Building Data Structures	91
5.2.3	Standard Memory Management	93
	<u>new</u>	93
	<u>dispose</u>	94
6.	Space Allocation for Data Types	95
6.1	Packed Data	95
6.2	Simple Types	97
6.2.1	Integer	97
6.2.2	Real	97
6.2.3	Long Integer	98
6.2.4	Boolean	98
6.2.5	Character	98
6.3	Scalars and Subranges	99
6.4	Structured Types	99
6.4.1	Arrays	99
6.4.2	Strings	99
6.4.3	Sets	99
6.4.4	Records	100
6.5	Dynamic Types	100
6.5.1	Files	100
6.5.2	Pointers	101
6.6	The Sizeof Intrinsic	101

IV. Overall Program Syntax	103
1. The Outline of a Program	103
1.1 Program Heading	104
1.2 Label Declarations	104
1.3 Constant Declarations	104
1.4 Type Declarations	105
1.5 Variable Declarations	106
1.6 Routine Declarations	106
1.7 The Main Body	107
2. Structure and Scope	107
2.1 Scope	107
3. Restrictions	109
3.1 Forward Declarations	109
3.2 Size Limits	110
3.3 Segment Routines	110
4. Units and Separate Compilation	111
4.1 The Uses Declaration	111
4.2 The Format of a Unit	112
4.2.1 Restrictions	113
4.2.2 Initialization and Termination Code	114
4.3 The Implementation of Units	115
4.3.1 Previous Versions	115
V. Procedures and Functions	117
1. Procedure and Function Declarations	117
1.1 Value Parameters	119
1.2 Variable Parameters	119
1.3 Procedures and Functions as Parameters	120
2. Calling Procedures and Functions	120
VI. Control Statements	123
1. Compound Statements	123
2. Conditional Statements	124
2.1 The If Statement	124
2.2 The Case Statement	125
3. Repetition	127
3.1 The While Statement	128
3.2 The Repeat Statement	129
3.3 The For Statement	130
4. Branching	131
4.1 The Goto Statement	131
4.2 The Procedure Exit	132
4.3 The Procedure Halt	133

VII. Input and Output	135
1. Standard Pascal I/O	135
1.1 Read and Readln	135
1.2 Write and Writeln	137
1.2.1 Field Specifications	138
1.3 Eof and Eoln	139
1.4 Page	139
2. Handling External Files	140
2.1 Opening and Closing Files	140
<u>reset</u>	140
<u>rewrite</u>	141
<u>close</u>	142
2.2 Random Access to Files	142
<u>seek</u>	143
2.3 Untyped Files	143
<u>blockread</u>	143
<u>blockwrite</u>	144
3. The UCSD p-System Environment	146
3.1 Keyboard	146
3.2 Device I/O	146
<u>unitbusy</u>	147
<u>unitclear</u>	147
<u>unitread</u>	147
<u>unitstatus</u>	148
<u>unitwait</u>	149
<u>unitwrite</u>	149
3.3 The Time Procedure	150
3.4 Ioresult	151
3.5 Screen I/O	152
<u>gotoxy</u>	152
VIII. Memory Management	153
1. The p-System Runtime Environment	153
2. Segmentation	154
2.1 Code Segments	154
2.2 Controlling Segment Residence	155
<u>memlock</u>	155
<u>memswap</u>	155
3. Free Space	156
<u>memavail</u>	156
<u>varavail</u>	156
4. Free Space on the Heap	157
<u>varnew</u>	157
<u>vardispose</u>	158
<u>mark</u>	158
<u>release</u>	158

IX. Concurrency	159
1. Concurrent Execution	159
2. Processes	159
3. Initiating a Process	161
<u>start</u>	161
4. Process Synchronization	162
4.1 Semaphores	162
<u>seminit</u>	163
4.2 Signal and Wait	163
4.3 Interrupts	163
<u>attach</u>	163
X. Compilation	165
1. Compiler Options	165
1.1 Stack Options	165
1.2 Switch Options	166
1.3 String Options	168
1.4 Conditional Compilation	170
2. External Routines	172
 PART II UCSD Pascal/A Guide for Programmers	
0. Introduction	175
1. Bootstrapping the Programmer	177
Program 1: Factorials	177
Program 2: Prime Factors	184
Program 3: Pascal's Triangle	189
Program 4: Palindromes (procedure)	195
Program 5: Palindromes (function)	198
Program 6: Roman Numerals	204
2. Data and Expressions	209
Program 7: Long Integer Factorials	215
Program 8: Quadratic Equation	217
Program 9: Uppercase Translation	220
Program 10: Matrix Multiplication	224
Program 11: Integer/String Conversion	227
Program 12: Pos Function	229
Program 13: Uppercase (with sets)	231
Program 14: Command Prompts	232
Program 15: Date Increment	235
Program 16: Symbol Table Routines	239