

Borland's No-Nonsense License Statement!

This software is protected by both United States Copyright Law and International Treaty provisions. Therefore, you must treat this software *just like a book* with the following single exception: Borland International authorizes you to make archival copies of the software for the sole purpose of backing up your software and protecting your investment from loss.

By saying, "just like a book," Borland means, for example, that this software may be used by any number of people and may be freely moved from one computer location to another so long as there is **No Possibility** of it being used at one location while it's being used at another. Just like a book that can't be read by two different people in two different places at the same time, neither can the software be used by two different people in two different places at the same time. (Unless, of course, Borland's Copyright has been violated.)

WARRANTY

With respect to the physical diskette and physical documentation enclosed herein, BORLAND INTERNATIONAL, INC. ("BORLAND") warrants the same to be free of defects in materials and workmanship for a period of 60 days from the date of purchase. In the event of notification within the warranty period of defects in material or workmanship, BORLAND will replace the defective diskette or documentation. The remedy for breach of this warranty shall be limited to replacement and shall not encompass any other damages, including but not limited to loss of profit, special, incidental, consequential, or other similar claims.

BORLAND INTERNATIONAL, INC., SPECIFICALLY DISCLAIMS ALL OTHER WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE WITH RESPECT TO DEFECTS IN THE DISKETTE AND DOCUMENTATION, AND THE PROGRAM LICENSE GRANTED HEREIN. IN PARTICULAR, AND WITHOUT LIMITING OPERATION OF THE PROGRAM LICENSE WITH RESPECT TO ANY PARTICULAR APPLICATION, USE, OR PURPOSE. IN NO EVENT SHALL BORLAND BE LIABLE FOR ANY LOSS OF PROFIT OR ANY OTHER COMMERCIAL DAMAGE, INCLUDING BUT NOT LIMITED TO SPECIAL, INCIDENTAL, CONSEQUENTIAL OR OTHER DAMAGES.

GOVERNING LAW

This Statement shall be construed, interpreted and governed by the laws of the State of California.

First Edition, August 1985
Printed in the United States of America
9 8 7 6

Turbo Pascal

version 3.0

Reference Manual

Copyright ©1983
Copyright ©1984
Copyright ©1985
by

BORLAND INTERNATIONAL Inc.
4585 Scotts Valley Drive
Scotts Valley, CA 95066

TABLE OF CONTENTS

INTRODUCTION	1
The Pascal Language	1
TURBO Pascal	1
Structure of This Manual	2
Typography	4
Syntax Descriptions	4
Chapter 1. USING THE TURBO SYSTEM	7
.COM and .CMD files	7
BEFORE USE	7
IMPORTANT NOTE !!!	7
Files On The Distribution Disk	8
Starting TURBO Pascal	10
Installation	12
IBM PC Screen Installation	12
Non-IBM PC Screen Installation	12
Installation of Editing Commands	13
The Menu	14
Logged Drive Selection	15
Work File Selection	15
Main File Selection	16
Edit Command	17
Compile Command	17
Run Command	17
Save Command	17
Directory Command	18
Quit Command	18
compiler Options	18
The TURBO Editor	19
The Status Line	19
Editing Commands	20
A Note on Control Characters	22
Before You Start: How To Get Out	22
Basic Movement Commands	22
Extended Movement Commands	25
Insert and Delete Commands	26
Block Commands	28
Miscellaneous Editing Commands	30

The TURBO editor vs. WordStar	34	Chapter 6. EXPRESSIONS	51
Cursor Movement	34	Operators	51
Mark Single Word	34	Unary Minus	51
End Edit	35	Not Operator	52
Line Restore	35	Multiplying Operators	52
Tabulator	35	Adding Operators	53
Auto Indentation	35	Relational Operators	53
Chapter 2. BASIC LANGUAGE ELEMENTS	37	Function Designators	54
Basic Symbols	37	Chapter 7. STATEMENTS	55
Reserved Words	37	Simple Statements	55
Standard Identifiers	38	Assignment Statement	55
Delimiters	39	Procedure Statement	56
Program Lines	39	Goto Statement	56
Chapter 3. STANDARD SCALAR TYPES	41	Empty Statement	56
Integer	41	Structured Statements	57
Byte	41	Compound Statement	57
Real	42	Conditional Statements	57
Boolean	42	If Statement	57
Char	42	Case Statement	58
Chapter 4. USER DEFINED LANGUAGE ELEMENTS	43	Repetitive Statements	59
Identifiers	43	For Statement	60
Numbers	43	While statement	61
Strings	44	Repeat Statement	61
Control Characters	45	Chapter 8. SCALAR AND SUBRANGE TYPES	63
Comments	46	Scalar Type	63
Compiler Directives	46	Subrange Type	64
Chapter 5. PROGRAM HEADING AND PROGRAM BLOCK	47	Type Conversion	65
Program Heading	47	Range Checking	65
Declaration Part	47	Chapter 9. STRING TYPE	67
Label Declaration Part	48	String Type Definition	67
Constant Definition Part	48	String Expressions	67
Type Definition Part	49	String Assignment	68
Variable Declaration Part	49	String Procedures	69
Procedure and Function Declaration Part	50	Delete	69
Statement Part	50	Insert	69
		Str	70
		Val	70

String Functions	71
Copy	71
Concat	71
Length	72
Pos	72
Strings and Characters	73
Chapter 10. ARRAY TYPE	75
Array Definition	75
Multidimensional Arrays	76
Character Arrays	77
Predefined Arrays	77
Chapter 11. RECORD TYPE	79
Record Definition	79
With Statement	81
Variant Records	82
Chapter 12. SET TYPE	85
Set Type Definition	85
Set Expressions	86
Set Constructors	86
Set Operators	87
Set Assignments	88
Chapter 13. TYPED CONSTANTS	89
Unstructured Typed Constants	89
Structured Typed Constants	90
Array Constants	90
Multi-dimensional Array Constants	91
Record Constants	91
Set Constants	92
Chapter 14. FILE TYPES	93
File Type Definition	93
Operations on Files	94
Assign	94
Rewrite	94
Reset	94
Read	95
Write	95
Seek	95
Flush	96
Close	96

Erase	96
Rename	96
File Standard Functions	97
EOF	97
FilePos	97
FileSize	98
Using Files	98
Text Files	101
Operations on Text Files	101
ReadLn	101
WriteLn	101
Eoln	102
SeekEoln	102
SeekEof	102
Logical Devices	104
Standard Files	105
Text Input and Output	108
Read Procedure	108
ReadLn Procedure	110
Write Procedure	111
Write Parameters	112
Writeln Procedure	113
Untyped Files	114
BlockRead / BlockWrite	114
I/O checking	116
Chapter 15. POINTER TYPES	119
Defining a Pointer Variable	119
Allocating Variables (New)	120
Mark and Release	120
Using Pointers	122
Dispose	124
GetMem	125
FreeMem	125
MaxAvail	126

Chapter 16. PROCEDURES AND FUNCTIONS	127
Parameters	127
Relaxations on Parameter Type Checking	129
Untyped Variable Parameters	130
Procedures	131
Procedure Declaration	131
Standard Procedures	133
ClrEol	133
ClrScr	133
CrtInit	133
CrtExit	134
Delay	134
DelLine	134
InsLine	134
GotoXY	134
Exit	135
Halt	135
LowVideo	135
NormVideo	135
Randomize	135
Move	136
FillChar	136
Functions	137
Function Declaration	137
Standard Functions	139
Arithmetic Functions	139
Abs	139
ArcTan	139
Cos	139
Exp	140
Frac	140
Int	140
Ln	140
Sin	140
Sqr	141
Sqrt	141
Scalar Functions	141
Pred	141
Succ	141
Odd	141

Transfer Functions	142
Chr	142
Ord	142
Round	142
Trunc	142
Miscellaneous Standard Functions	143
Hi	143
KeyPressed	143
Lo	143
Random	143
Random(Num)	143
SizeOf	144
Swap	144
UpCase	144
Forward References	145
Chapter 17. INCLUDING FILES	147
Chapter 18. OVERLAY SYSTEM	149
Creating Overlays	152
Nested Overlays	154
Automatic Overlay Management	155
Placing Overlay Files	155
Efficient Use of Overlays	155
Restrictions Imposed on Overlays	156
Data Area	156
Forward Declarations	156
Recursion	156
Run-Time Errors	156
Chapter 19. IBM PC GOODIES	159
Screen Mode Control	160
Text Modes	160
Color Modes	161
TextColor	161
TextBackground	162
Cursor Position	162
WhereX	162
WhereY	162

Graphics Modes	163
GraphColorMode	163
GraphMode	164
HiRes	164
HiResColor	164
Palette	165
GraphBackground	166
Windows	168
Text Windows	168
Graphics Windows	169
Basic Graphics	171
Plot	171
Draw	171
Extended Graphics	172
ColorTable	172
Arc	173
Circle	173
GetPic	173
PutPic	174
GetDotColor	174
FillScreen	175
FillShape Procedure	175
FillPattern	175
Pattern	176
Turtlegraphics	177
Back	178
ClearScreen	179
Forward	179
Heading	179
HideTurtle	179
Home	179
NoWrap	180
PenDown	180
PenUp	180
SetHeading	180
SetPenColor	181
SetPosition	181
ShowTurtle	181
TurnLeft	181
TurnRight	181
TurtleWindow	182
TurtleThere	183
TurtleDelay	183
Wrap	184
Xcor	184

Ycor	184
Sound	185
Editor Command Keys	186
Chapter 20. PC-DOS AND MS-DOS	187
Tree-Structured Directories	187
On the Main Menu	187
Directory-related procedures	189
ChDir	189
MkDir	189
Rmdir	189
GetDir	189
Compiler Options	190
Memory / Com file / cHn-file	190
Minimum Code Segment Size	191
Minimum Data Segment Size	191
Minimum Free Dynamic Memory	192
Maximum Free Dynamic Memory	192
Command Line Parameter	192
Find Run-time Error	192
Standard Identifiers	193
Chain and Execute	193
Overlays	196
OvrPath Procedure	196
Files	198
File Names	198
Number of Open Files	198
Extended File Size	199
File of Byte	199
Flush Procedure	199
Truncate Procedure	199
Text Files	200
Buffer Size	200
Append Procedure	200
Flush Procedure	200
Logical Devices	200
I/O redirection	201
Absolute Variables	203
Absolute Address Functions	204
Addr	204
Ofs	204
Seg	204
Cseg	205
Dseg	205
Sseg	205

Predefined Arrays	205
Mem Array	205
Port Array	206
With Statements	206
Pointer Related Items	206
MemAvail	206
Pointer Values	207
Assigning a Value to a Pointer	207
Obtaining The Value of a Pointer	207
DOS Function Calls	208
User Written I/O Drivers	209
External Subprograms	210
In-line Machine Code	211
Interrupt Handling	214
Intr procedure	214
Internal Data Formats	216
Basic Data Types	216
Scalars	216
Reals	217
Strings	217
Sets	218
Pointers	218
Data Structures	219
Arrays	219
Records	219
Disk Files	220
File Interface Blocks	220
Random Access Files	221
Text Files	221
Parameters	221
Variable Parameters	223
Value Parameters	223
Scalars	223
Reals	223
Strings	223
Sets	224
Pointers	224
Arrays and Records	224
Function Results	224
The Heap and The Stacks	225
Memory Management	226

Chapter 21. CP/M-86	227
Compiler Options	227
Memory / Cmd file / cHn-file	227
Minimum Code Segment Size	228
Minimum Data Segment Size	229
Minimum Free Dynamic Memory	229
Maximum Free Dynamic Memory	229
Command Line Parameter	229
Find Runtime Error	229
Standard Identifiers	230
Chain and Execute	231
Overlays	233
OvrDrive Procedure	233
Files	235
File Names	235
Untyped files	235
Text Files	235
Buffer Size	235
Absolute Variables	236
Absolute Address Functions	237
Addr	237
Ofs	237
Seg	237
Cseg	237
Dseg	238
Sseg	238
Predefined Arrays	238
Mem Array	238
Port Array	239
With Statements	239
Pointer Related Items	239
MemAvail	239
Pointer Values	239
Assigning a Value to a Pointer	240
Obtaining The Value of a Pointer	240
CP/M-86 Function Calls	240
User Written I/O Drivers	241
External Subprograms	242
In-line Machine Code	243
Interrupt Handling	245
Intr procedure	245

Internal Data Formats	246
Basic Data Types	246
Scalars	247
Reals	247
Strings	248
Sets	248
Pointers	249
Data Structures	249
Arrays	249
Records	250
Disk Files	250
File Interface Blocks	250
Random Access Files	251
Text Files	252
Parameters	252
Variable Parameters	253
Value Parameters	253
Scalars	254
Reals	254
Strings	254
Sets	254
Pointers	254
Arrays and Records	254
Function Results	255
The Heap and The Stacks	255
Memory Management	256
Chapter 22. CP/M-80	259
eXecute Command	259
compiler Options	259
Memory / Com file / cHn-file	260
Start Address	261
End Address	261
Command Line Parameter	262
Find Runtime Error	262
Standard Identifiers	263
Chain and Execute	263
Overlays	265
OvrDrive Procedure	265
Files	267
File Names	267
Text Files	267
Absolute Variables	267
Addr Function	268

Predefined Arrays	268
Mem Array	268
Port Array	269
Array Subscript Optimization	269
With Statements	269
Pointer Related Items	270
MemAvail	270
Pointers and Integers	270
CP/M Function Calls	271
Bdos procedure and function	271
BdosHL function	271
Bios procedure and function	272
BiosHL function	272
User Written I/O Drivers	272
External Subprograms	274
In-line Machine Code	274
Interrupt Handling	277
Internal Data Formats	278
Basic Data Types	278
Scalars	278
Reals	278
Strings	279
Sets	279
File Interface Blocks	280
Pointers	281
Data Structures	281
Arrays	281
Records	282
Disk Files	282
Random Access Files	282
Text Files	283
Parameters	283
Variable Parameters	283
Value Parameters	283
Scalars	283
Reals	284
Strings	284
Sets	284
Pointers	285
Arrays and Records	285
Function Results	285
The Heap and The Stacks	286

Memory Management	288
Memory Maps	288
Compilation in Memory	288
Compilation To Disk	289
Execution in Memory	290
Execution of A Program File	291
Chapter 23. TURBO BCD PASCAL	293
Files On the TURBO BCD Pascal Distribution Diskette	293
BCD Range	293
Form function	294
Numeric Fields	294
String Fields	297
Writing BCD Reals	297
Formatted Writing	298
Internal Data Format	298
Chapter 24. TURBO-87	301
Files On the TURBO-87 Distribution Diskette	301
Writing 8087 Reals	302
Internal Data Format	302
Appendix A. SUMMARY OF STANDARD PROCEDURES AND FUNCTIONS	303
Input/Output Procedures and Functions	303
Arithmetic Functions	304
Scalar Functions	304
Transfer Functions	304
String Procedures and Functions	305
File Handling Routines	305
Heap Control Procedures and Functions	306
Screen Related Procedures and Functions	306
Miscellaneous Procedures and Functions	307
IBM PC Procedures and Functions	308
Basic Graphics, Windows, and Sound	308
Extended Graphics	309
Turtlegraphics	309
Appendix B. SUMMARY OF OPERATORS	311

Appendix C. SUMMARY OF COMPILER DIRECTIVES	313
IMPORTANT NOTICE	313
Common Compiler Directives	314
B - I/O Mode Selection	314
C - Control C and S	314
I - I/O Error Handling	314
I - Include Files	314
R - Index Range Check	315
V - Var-parameter Type Checking	315
U - User Interrupt	315
PC-DOS and MS-DOS Compiler Directives	316
G - Input File Buffer	316
P - Output File Buffer	316
D - Device Checking	316
F - Number of Open Files	317
PC-DOS, MS-DOS, and CP/M-86 Compiler Directive	317
K - Stack Checking	317
CP/M-80 Compiler Directives	318
A - Absolute Code	318
W - Nesting of With Statements	318
X - Array Optimization	318
Appendix D. TURBO VS. STANDARD PASCAL	319
Dynamic Variables	319
Recursion	319
Get and Put	319
Goto Statements	319
Page Procedure	320
Packed Variables	320
Procedural Parameters	320
Appendix E. COMPILER ERROR MESSAGES	321
Appendix F. RUN-TIME ERROR MESSAGES	325
Appendix G. I/O ERROR MESSAGES	327
Appendix H. TRANSLATING ERROR MESSAGES	329
Error Message File Listing	330
Appendix I. TURBO SYNTAX	333

Appendix J. ASCII TABLE	339
Appendix K. KEYBOARD RETURN CODES	341
Appendix L. INSTALLATION	345
Terminal Installation	345
IBM PC Display Selection	345
Non-IBM PC Installation	346
Editing Command Installation	350
Appendix M. CP/M PRIMER	355
How to use TURBO on a CP/M system	355
Copying Your TURBO Disk	355
Using Your TURBO Disk	356
Appendix N. HELP!!!	357
Appendix O. SUBJECT INDEX	363

LIST OF FIGURES

1-1 Log-on Message	10
1-2 Main Menu	11
1-3 Installation Main Menu	12
1-4 Main Menu	14
1-5 Editor Status Line	19
15-1 Using Dispose	124
18-1 Principle of Overlay System	149
18-2 Largest Overlay Subprogram Loaded	150
18-3 Smaller Overlay Subprogram Loaded	151
18-4 Multiple Overlay Files	153
18-5 Nested Overlay Files	154
19-1 Text Windows	169
19-2 Graphics Windows	170
19-3 Turtle Coordinates	178
19-4 Turtle Coordinates	183
20-1 TURBO Main Menu under DOS 2.0	187
20-2 Options Menu	190
20-3 Memory Usage Menu	191
20-4 Run-time Error Message	192
20-5 Find Run-time Error	192
21-1 Options Menu	227
21-2 Memory Usage Menu	228
21-3 Run-time Error Message	230
21-4 Find Run-time Error	230
22-1 Options Menu	260
22-2 Start and End Addresses	261
22-3 Run-time Error Message	262
22-4 Find Run-time Error	262
22-5 Memory map during compilation in memory	288
22-6 Memory map during compilation to a file	289
22-7 Memory map during execution in direct mode	290
22-8 Memory map during execution of a program file	291
L-1 IBM PC Screen Installation Menu	345
L-2 Terminal Installation Menu	346

LIST OF TABLES

1-1 Editing Command Overview	21
14-1 Operation of EOLN and Eof	105
19-1 Text Mode Color Scale	161
19-2 High Resolution Graphics Color Scale	165
19-3 Color Palettes in Color Graphics	165
19-4 Color Palettes in B/W Graphics	166
19-5 Graphics Background Color Scale	167
19-6 IBM PC Keyboard Editing Keys	186
K-1 Keyboard Return Codes	343
L-1 Secondary Editing Commands	353

INTRODUCTION

This book is a reference manual for the TURBO Pascal system as implemented for the PC-DOS, MS-DOS, CP/M-86, and CP/M-80 operating systems. Although making thorough use of examples, it is not meant as a Pascal tutorial or textbook, and at least a basic knowledge of Pascal is assumed.

A TURBO Pascal Tutorial, however, is also available from Borland. Please see page 3 for ordering information.

The Pascal Language

Pascal is a general-purpose, high level programming language originally designed by Professor Niklaus Wirth of the Technical University of Zurich, Switzerland and named in honor of Blaise Pascal, the famous French Seventeenth Century philosopher and mathematician.

Professor Wirth's definition of the Pascal language, published in 1971, was intended to aid the teaching of a systematic approach to computer programming, specifically introducing *structured programming*. Pascal has since been used to program almost any task on almost any computer and it is today established as one of the foremost high-level languages; whether the application is education, hobby, or professional programming.

TURBO Pascal

TURBO Pascal is designed to meet the requirements of all categories of users: it offers the student a friendly interactive environment which greatly aids the learning process; and in the hands of a programmer it becomes an extremely effective development tool providing both compilation and execution times second to none.

TURBO Pascal closely follows the definition of Standard Pascal as defined by K. Jensen and N. Wirth in the *Pascal User Manual and Report*. The few and minor differences are described in Appendix D. In addition to the standard, a number of extensions are provided, such as:

Absolute address variables
Bit/byte manipulation
Direct access to CPU memory and data ports
Dynamic strings
Free ordering of sections within declaration part
Full support of operating system facilities
In-line machine code generation
Include files
Logical operations on integers
Overlay system
Program chaining with common variables
Random access data files
Structured constants
Type conversion functions

IBM PC and compatibles only:

Colors
Graphics
Turtlegraphics
Windows
Sound

Furthermore, many extra standard procedures and functions are included to increase the versatility of TURBO Pascal.

Structure of This Manual

As this manual describes slightly different TURBO Pascal implementations, namely PC-DOS, MS-DOS, CP/M-86, and CP/M-80, the reader should keep the following structure in mind:

- 1: Chapter 1 describes the installation and use of TURBO Pascal, the built-in editor, etc. This information applies to all implementations.
- 2: The main body of the manual, chapters 2 through 18, describe the common parts of TURBO Pascal, i.e. those parts of the language which are identical in all three versions. These include Standard Pascal and many extensions. As long as you use the language as described in these chapters, your programs will be fully portable between implementations.

- 3: Chapters 19, 20, 21, and 22 describe items which have not been covered in previous chapters because they differ among implementations, for example special features, requirements, and limitations of each implementation. In particular, you should notice that chapter 19 explains all the IBM PC extensions such as colors, graphics, sound, windows, etc. To avoid confusion, you need only read the chapter(s) pertaining to your implementation.

Parts of chapters 20, 21, and 22 deal with technicalities such as internal data formats, interrupts, direct memory and port accesses, in-line assembly code, user written I/O drivers, etc. **It is assumed that the reader has previous knowledge of such matters, and no attempt is made to teach these things.** Remember that these chapters are implementation dependent, so programs using techniques described there are no longer directly portable between implementations.

In fact, you need not bother with these chapters at all if your aim is to write plain Pascal code, or if portability between the different TURBO implementations is important to you.

- 4: Chapter 23 describes TURBO-BCD. This is a special version of TURBO Pascal for PC-DOS, MS-DOS, and CP/M-86 which uses binary coded decimal (BCD) arithmetic for higher precision in real operations; especially useful for business applications.
- 5: Chapter 24 describes the special 16-bit TURBO-87 which uses the optional 8087 co-processor for added speed and extended range in *Real* arithmetic.
- 6: The appendices are common to all implementations and contain summaries of language elements, syntax diagrams, error messages, details on installation procedures, an alphabetical subject index, etc.
- 7: Appendix N contains answers to a number of the most common questions—please read it if you have any problems.

TURBO Pascal equipped with either BCD or 8087 options is available for an additional fee at better dealers nationwide. Call (800) 556-2283 for the dealer nearest you. To order by credit card, call (800) 255-8008, in California call (800) 742-1133.

Typography

The body of this manual is printed in normal typeface. Special characters are used for the following special purposes:

Typewriter

Typewriter-characters are used to illustrate program examples and screen output. Screen images are furthermore shown in rectangular fields of thin lines.

Italics

Italics are used in general to emphasize sections of the text. In particular, pre-defined standard identifiers and elements in syntax descriptions (see below) are printed in italics. The meaning of the use of italics thus depends on the context.

Boldface

Boldface is used to mark reserved words; and also to highlight particularly important passages in the text.

Syntax Descriptions

The entire syntax of the Pascal language expressed as *Backus-Naur Forms* is collected in Appendix I which also describes the typography and special symbols used in these forms.

Where appropriate syntax descriptions are also used more specifically to show the syntax of single language elements as in the following syntax description of the function *Concat*:

```
Concat ( St1 , St2 ( , StN ) )
```

Reserved words are printed in **boldface**, identifiers use mixed upper and lower case, and elements explained in the text are printed in *italics*.

The text will explain that *St1*, *St2*, and *StN* must be string expressions. The syntax description shows that the word *Concat* must be followed by two or more string expressions, separated by commas and enclosed in parentheses. In other words, the following examples are legal (assuming that *Name* is a string variable):

```
Concat('TURBO', 'Pascal')  
Concat('TU', 'RB0', 'Pascal')  
Concat('T', 'U', 'R', 'B', '0', Name)
```