

# Contents

	<b>Preface</b> . . . . .	<b>xv</b>
<b>Chapter 1:</b>	<b>An Introduction to Functions.</b> . . . . .	<b>1</b>
	What Is a Function? . . . . .	1
	Functions vs. Operation Codes . . . . .	2
	Return Values . . . . .	3
	Function Parameters . . . . .	4
	Summary . . . . .	6
<b>Chapter 2:</b>	<b>Essential BIFs in Free-Format RPG IV</b> . . . . .	<b>7</b>
	Different Uses for Built-in Functions . . . . .	7
	BIFs Used in Definition Specifications . . . . .	8
	The %size BIF . . . . .	8
	The %len BIF. . . . .	8
	The %decpos BIF. . . . .	9
	The %addr BIF. . . . .	9
	The %paddr BIF . . . . .	9
	The %elem BIF . . . . .	10
	BIFs in a Procedure, by Return Value . . . . .	10
	The Indicator Return Data Type . . . . .	10
	The Numeric Return Data Type . . . . .	12
	The Date, Time, and Timestamp Return Data Types. . . . .	14
	The Character Return Data Type . . . . .	15
	The Pointer Return Data Type. . . . .	16
	BIFs with No Return Values . . . . .	17
	Summary . . . . .	17
<b>Chapter 3:</b>	<b>Using C Functions</b> . . . . .	<b>19</b>
	Why Use C Functions? . . . . .	19
	How C Functions Work with RPG IV . . . . .	21

Random Numbers from C . . . . .	22
C Data Types vs. RPG IV Data Types . . . . .	24
Parameter Passing to C Functions. . . . .	25
Character String Differences between C and RPG IV . . . . .	26
Binding RPG IV and C Functions . . . . .	26
Using C Functions to Make Your Job Easier . . . . .	27
The Strtok Function . . . . .	28
Exponentiation . . . . .	30
Summary . . . . .	31

<b>Chapter 4: Subprocedures . . . . .</b>	<b>33</b>
Comparing Subprocedures and Subroutines . . . . .	33
Comparing Subprocedures and Static-bound Modules . . . . .	34
The Anatomy of a Subprocedure . . . . .	34
A Typecheck Subprocedure. . . . .	35
A Date-conversion Subprocedure. . . . .	37
Subprocedures within a Source Member. . . . .	40
Subprocedures outside of a Source Member. . . . .	43
Summary . . . . .	47

<b>Chapter 5: Using Prototyping and ILE to Connect the Parts . . . . .</b>	<b>49</b>
Prototyping . . . . .	49
The Procedure Interface . . . . .	52
Prototype/Procedure Interface Keywords . . . . .	54
Value . . . . .	54
Const . . . . .	55
Options(*NoPass). . . . .	55
Options(*Omit) . . . . .	55
Options(*Varsize). . . . .	56
Options(*String) . . . . .	56
Options(*RightAdj) . . . . .	56
Options(*Trim) . . . . .	56
Binding to Make “Modular” Work Efficiently . . . . .	56
Bind-by-Copy . . . . .	57
Bind-by-Reference . . . . .	58
Comparing Bind-by-Copy with Bind-by-Reference . . . . .	60
The Service Program Signature . . . . .	60
The Binding Directory . . . . .	62
API Interfacing . . . . .	63
Dynamic Calls . . . . .	63
Activation Groups. . . . .	65
The Default Activation Group . . . . .	66
The Named Activation Group . . . . .	67

The Activation Group *New . . . . .	68
The Activation Group *Caller . . . . .	68
Scoping Resources . . . . .	68
Performance Considerations . . . . .	69
Cleaning up an Activation Group . . . . .	69
Displaying Activation Groups for a Job . . . . .	70
Summary . . . . .	70
<b>Chapter 6: Coding RPG IV the Modular and Functional Way . . . . .</b>	<b>71</b>
Analyze the Program Needs . . . . .	71
Gather and/or Code the Needed Subprocedures or Modules . . . . .	73
Use Nested BIFs to Eliminate Work Fields . . . . .	74
Summary . . . . .	76
<b>Appendix A: Free-format RPG IV Coding Standards and Recommendations . . . . .</b>	<b>77</b>
Standards . . . . .	77
Comments . . . . .	77
A Prologue . . . . .	78
Blank Lines to Separate Related Source Lines . . . . .	78
Comments at the End of a Line . . . . .	78
Code Changes Marked in Positions 1–5 . . . . .	78
Definition Specifications . . . . .	79
Naming Conventions . . . . .	79
Indicators . . . . .	80
Structured Programming Techniques . . . . .	80
Modular Programming Techniques . . . . .	80
Character String Manipulation . . . . .	81
The Operation Extender . . . . .	81
The %fields Option on the Update Operation . . . . .	81
Unsigned Integer Data Types . . . . .	81
Filenames on %eof and %found BIFs . . . . .	81
Miscellaneous . . . . .	82
Recommended Coding Practices . . . . .	82
The %equal BIF after a Setxx Operation . . . . .	82
Date and Time Data Types and Operations . . . . .	82
BIF Nesting . . . . .	82
Monitor/On-error/Endmon for Operations Where (e) Is Unavailable . . . . .	82
Programmer's Choice . . . . .	83
A Comment on the Same Line as an Operation . . . . .	83
Accumulation Operators . . . . .	83
Capitalizing the First Letter of an Operation . . . . .	83

Mixed Case within a Name . . . . .	83
Special Characters within a Name . . . . .	84

**Appendix B: Built-in Functions Reference . . . . . 85**

%abs . . . . .	85
%addr . . . . .	86
%alloc . . . . .	88
%bitand . . . . .	88
%bitnot . . . . .	90
%bitor . . . . .	91
%bitxor . . . . .	92
%char . . . . .	93
%check. . . . .	94
%checkr . . . . .	96
%date . . . . .	96
%dec, %dech. . . . .	98
%decpos . . . . .	100
%diff . . . . .	101
%div . . . . .	102
Duration BIFs . . . . .	103
%editc . . . . .	104
%editflt. . . . .	107
%editw . . . . .	108
%elem . . . . .	111
%eof . . . . .	112
%equal . . . . .	113
%error . . . . .	114
%fields . . . . .	115
%float. . . . .	116
%found . . . . .	117
%graphic . . . . .	118
%handler. . . . .	118
%int, %inth . . . . .	120
%kds . . . . .	121
%len . . . . .	123
%lookupxx . . . . .	124
%nullind . . . . .	127
%occur . . . . .	128
%open . . . . .	129
%paddr . . . . .	130
%parms. . . . .	131
%realloc . . . . .	131
%rem . . . . .	132

%replace . . . . .	134
%scan. . . . .	136
%shtdn . . . . .	137
%size . . . . .	138
%sqrt . . . . .	141
%status. . . . .	142
%str . . . . .	142
%subarr . . . . .	144
%subdt . . . . .	146
%subst . . . . .	148
%time. . . . .	149
%timestamp . . . . .	151
%tlookupxx. . . . .	152
%trimx . . . . .	154
%ucs2. . . . .	155
%uns, %unsh. . . . .	156
%xfoot . . . . .	157
%xlate . . . . .	158
%xml . . . . .	159