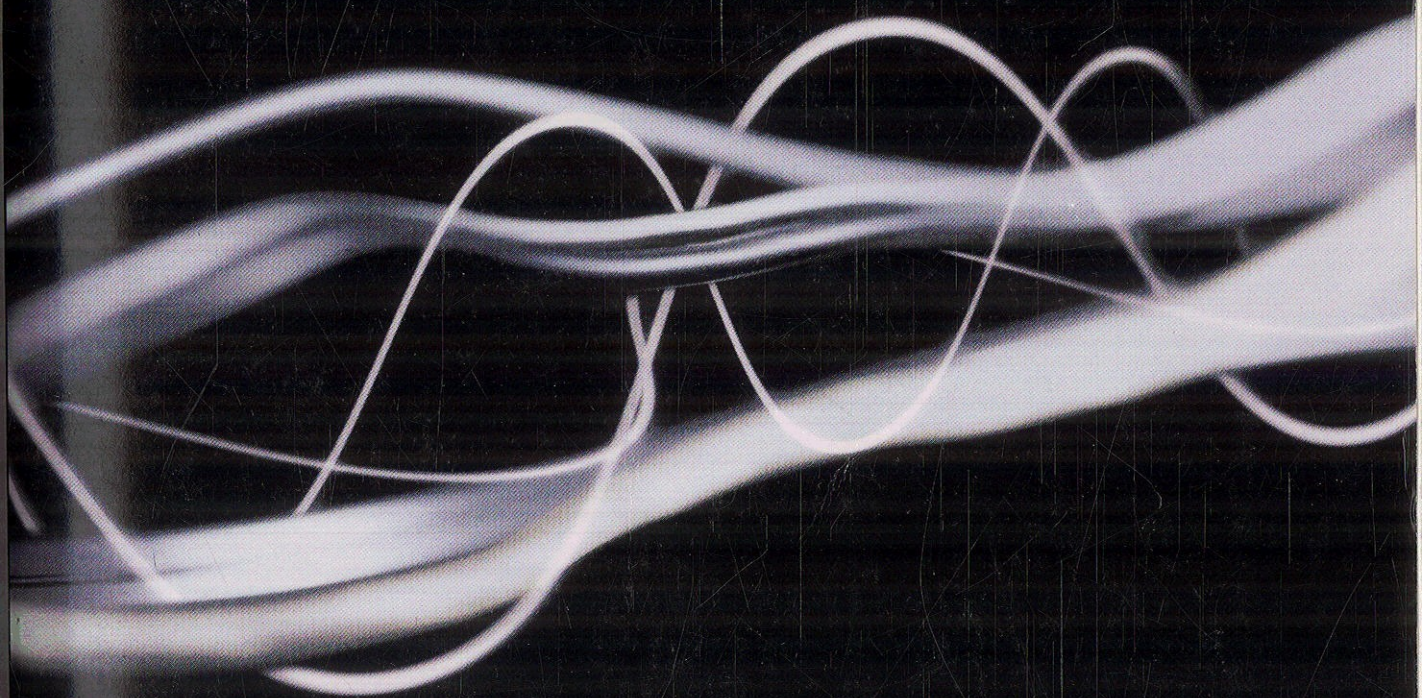


THIRD EDITION

Writing **Compilers and Interpreters** A Software Engineering Approach



Ronald Mak

Contents

Introduction	xxi
Chapter 1 Introduction	1
Goals and Approach	1
What Are Compilers and Interpreters?	2
Comparing Compilers and Interpreters	4
The Picture Gets a Bit Fuzzy	5
Why Study Compiler Writing?	6
Conceptual Design	6
Syntax and Semantics	8
Lexical, Syntax, and Semantic Analyses	10
Chapter 2 Framework I: Compiler and Interpreter	11
Goals and Approach	11
Language-Independent Framework Components	12
Front End	13
Parser	16
Intermediate Tier	34
Back End	36
Pascal-Specific Front End Components	37
Pascal Parser	37
Pascal Scanner	39
A Front End Factory	41
Initial Back End Implementations	43
Interpreter	44
A Back End Factory	45
Program 2: Program Listings	46
Chapter 3 Scanning	55
Goals and Approach	55
Program 3: Pascal Tokenizer	57

	Syntax Error Handling	65
	How to Scan for Tokens	72
	A Pascal Scanner	75
	Pascal Tokens	77
	Syntax Diagrams	80
	Word Tokens	81
	String Tokens	82
	Special Symbol Tokens	85
	Number Tokens	88
Chapter 4	The Symbol Table	97
	Goals and Approach	97
	Symbol Table Conceptual Design	98
	The Symbol Table Stack	98
	Symbol Table Interfaces	100
	A Symbol Table Factory	105
	Symbol Table Implementation	107
	Program 4: Pascal Cross-Referencer I	113
Chapter 5	Parsing Expressions and Assignment Statements	121
	Goals and Approach	121
	Syntax Diagrams	122
	Intermediate Code Conceptual Design	125
	Intermediate Code Interfaces	126
	An Intermediate Code Factory	129
	Intermediate Code Implementation	130
	Printing Parse Trees	135
	Parsing Pascal Statements and Expressions	141
	Parsing Statements	145
	Parsing the Compound Statement	148
	Parsing the Assignment Statement	149
	Parsing Expressions	151
	Program 5: Pascal Syntax Checker I	161
Chapter 6	Interpreting Expressions and Assignment Statements	167
	Goals and Approach	167
	Runtime Error Handling	168
	Executing Assignment Statements and Expressions	170
	The Statement Executor Subclasses	170
	Executing Statements	173
	Executing the Compound Statement	175
	Executing the Assignment Statement	175
	Executing Expressions	177
	Program 6: Simple Interpreter I	184
Chapter 7	Parsing Control Statements	189
	Goals and Approach	189
	Syntax Diagrams	190

Error Recovery	191
Program 7: Syntax Checker II	192
Control Statement Parsers	193
Parsing Pascal Control Statements	198
Parsing the REPEAT Statement	198
Parsing the WHILE Statement	202
Parsing the FOR Statement	207
Parsing the IF Statement	214
Parsing the CASE Statement	219
Chapter 8 Interpreting Control Statements	233
Goals and Approach	233
Program 8: Simple Interpreter II	233
Interpreting Control Statements	234
Executing a Looping Statement	236
Executing the IF Statement	240
Executing the SELECT Statement	241
An Optimized SELECT Executor	245
Chapter 9 Parsing Declarations	249
Goals and Approach	249
Pascal Declarations	250
Types and the Symbol Table	253
Type Specification Interfaces	253
Pascal Type Specification Implementation	255
A Type Factory	260
Scope and the Symbol Table Stack	261
How Identifiers Are Defined	266
Predefined Types and Constants	268
Parsing Pascal Declarations	271
Parsing Constant Definitions	277
Parsing Type Definitions and Type Specifications	283
Parsing Variable Declarations	301
Program 9: Pascal Cross-Referencer II	306
Chapter 10 Type Checking	331
Goals and Approach	331
Type Checking	331
Type Checking Expressions	335
Type Checking Control Statements	350
Program 10: Pascal Syntax Checker III	358
Chapter 11 Parsing Programs, Procedures, and Functions	371
Goals and Approach	371
Program, Procedure, and Function Declarations	372
Nested Scopes and the Symbol Table Stack	375
New Declarations Parser Subclasses	378
Parsing a Program Declaration	382

Parsing Procedure and Function Declarations	382
Formal Parameter Lists	388
Parsing Procedure and Function Calls	394
Calls to Declared Procedures and Functions	398
Calls to the Standard Procedures and Functions	400
Actual Parameter Lists	408
Program 11: Pascal Syntax Checker IV	418
Chapter 12 Interpreting Pascal Programs	431
Goals and Approach	431
Runtime Memory Management	432
The Runtime Stack and Activation Records	432
The Runtime Display	436
Recursive Calls	439
Memory Management Interfaces and Implementation	440
The Memory Factory	459
Executing Statements and Expressions	460
The Executor Superclass	461
The Statement Executor	462
The Assignment and Expression Executors	469
Executing Procedure and Function Calls	478
Parameter Passing	478
Calling Procedures and Functions	478
Program 12-1: Pascal Interpreter	493
Chapter 13 An Interactive Source-Level Debugger	501
Goals and Approach	501
Machine-Level vs. Source-Level Debugging	502
Debugger Architecture	503
Runtime Data Input vs. Debugger Command Input	514
Creating a Command-Line Debugger	516
A Simple Command Language	517
Displaying Values	522
Parsing Variable Names	525
Executing Commands	529
Program 13-1: Command-Line Source-Level Debugger	540
Chapter 14 Framework II: An Integrated Development Environment (IDE)	543
Goals and Approach	544
The IDE Window	544
The Edit Window	545
The Debug Window	545
The Call Stack Window	547
The Console Window	548
Program 14: Pascal IDE	548

The IDE Process and the Debugger Process	549
The IDE Framework	549
Interprocess Communication	550
The Control Interface	560
The Debugger Process	566
Chapter 15 Jasmin Assembly Language and Code Generation for the Java Virtual Machine	577
Goals and Approach	577
Organization of the Java Virtual Machine	578
The Class Area	578
The Heap Area	579
The Java Runtime Stack	579
JVM Limitations	580
The Jasmin Assembly Language	581
Assembly Statements	581
Program Structure	593
Emitting Instructions	604
Load and Store Instructions	607
Data Manipulation Instructions	617
Control Instructions	620
Remaining Code Generation Methods	623
Chapter 16 Compiling Programs, Assignment Statements, and Expressions	625
Goals and Approach	625
Compiling Programs	626
Program Header	627
Class Constructor	627
Fields	627
Main Method	628
Code Generator Subclasses	629
Compiling Procedures and Functions	635
Parser and Symbol Table Changes	637
Generating Code for Procedures and Functions	639
Compiling Assignment Statements and Expressions	643
The Statement Code Generator	643
The Compound Statement Code Generator	645
The Assignment Statement Code Generator	646
The Expression Code Generator	648
The Pascal Runtime Library	655
Range Checking	655
Pascal Text Input	656
Building the Library	657
Program 16-1: Pascal Compiler I	657

Chapter 17	Compiling Procedure and Function Calls and String Operations	661
	Goals and Approach	661
	Compiling Procedure and Function Calls	662
	Value Parameters and VAR Parameters	664
	Calls to Declared Procedures and Functions	666
	Calls to the Standard Procedures and Functions	678
	The Pascal Runtime Library	691
	Pascal Input Text	692
	Building the Library	697
	Compiling Strings and String Assignments	698
	Allocating String Variables	698
	String Assignments	701
	String Comparisons	705
	Program 17-1: Pascal Compiler II	711
Chapter 18	Compiling Control Statements, Arrays, and Records	719
	Goals and Approach	719
	Compiling Control Statements	719
	Looping Statements	720
	The IF Statement	730
	The SELECT Statement	735
	Compiling Arrays and Subscripted Variables	744
	Allocating Memory for Arrays	744
	Subscripted Variables in Expressions and Assignments	757
	Compiling Records and Record Fields	767
	Allocating Records	768
	Record Fields in Expressions and Assignments	772
	Program 18-1: Pascal Compiler III	777
Chapter 19	Additional Topics	791
	Scanning	791
	Deterministic Finite Automata (DFA)	791
	Table-Driven Scanners	793
	Syntax Notation	796
	Backus-Naur Form (BNF)	796
	Extended BNF (EBNF)	797
	Grammars and Languages	797
	Parsing	798
	Top-Down Parsers	798
	Bottom-Up Parsers	798
	Context-Free and Context-Sensitive Grammars	800
	Code Generation	800
	Instruction Selection	800
	Instruction Scheduling	801
	Register Allocation	803
	Code Optimization	803

Debugging Compilers and Optimizing Compilers	804
Speed Optimizations	804
Runtime Memory Management	807
Heap and Stack	807
Garbage Collection	808
Compiling Object-Oriented Languages	809
Method Overloading and Name Mangling	809
Inheritance	810
Virtual Methods	810
Compiler-Compilers	811
JavaCC	811
Lex and Yacc	813

Index	817
--------------	------------