CAMBRIDGE
Cambridge University Press & Assessment
978-1-009-76751-4 — Control Structures for Programming Languages
Xavier Leroy
Table of Contents
More Information

Contents

Preface *page* ix
Introduction xi

Part I Control Structures for Imperative Languages 1

1 Early Programming Languages 3
1.1 Control flow in processors 3
1.2 Machine languages, assembly languages 4
1.3 FORTRAN 7
1.4 ALGOL 60 9
1.5 Conditionals and loops 11
1.6 Early exit from loops and blocks 13
1.7 Further reading 14

2 Structured Programming 15
2.1 Structured programming: a movement and a controversy 15
2.2 Programming without goto 17
2.3 Expressiveness of structured programming 19
2.4 Reductions between control structures 21
2.5 Reducible control-flow graphs 23
2.6 Reducibility and structured control 25
2.7 Further reading 28

3 Non-Local Control: Subroutines, Procedures, Functions 29
3.1 Subroutines for code reuse 29
3.2 Procedures and functions 30
3.3 Design dimensions and implementation strategies for functions 34
3.4 Control flow around function calls 38
3.5 Further reading 40

4 Control Inversion: Iterators, Generators, Coroutines 42
4.1 Iterators 42
4.2 Generators 45

© in this web service Cambridge University Press & Assessment www.cambridge.org

CAMBRIDGE
Cambridge University Press & Assessment
978-1-009-76751-4 — Control Structures for Programming Languages
Xavier Leroy
Table of Contents
More Information

4.3 Coroutines and cooperative threads 49
4.4 Further reading 51

Part II Control Operators for Functional Languages 53

5 Functional Languages 55
5.1 Declarative languages: getting rid of control? 55
5.2 XL: expressions and spreadsheets 56
5.3 APP: expressions and user-defined functions 60
5.4 FUN: functions as first-class values 62
5.5 Reductions under contexts 66
5.6 Further reading 69

6 Continuations and the CPS Transformation 70
6.1 Notion of continuation 70
6.2 Denotational semantics reminder 72
6.3 Continuation-based denotational semantics 73
6.4 Denotational semantics for labels and jumps 75
6.5 The CPS transformation 76
6.6 Semantic properties of the CPS transformation 79
6.7 Further reading 82

7 Programming with Continuations 83
7.1 Writing functions in CPS 83
7.2 Iterators 84
7.3 Asynchronous and concurrent programming 85
7.4 Errors and exceptions 86
7.5 Backtracking 87
7.6 Generating, filtering and counting 88
7.7 Further reading 90

8 Control Operators 91
8.1 Landin's J operator 91
8.2 Call-with-current-continuation (callcc) 92
8.3 Implementing control structures with callcc 94
8.4 Semantics of callcc 96
8.5 Delimited continuations 98
8.6 Semantics of delimited continuation operators 99
8.7 Implementing control structures with delimited continuations 101
8.8 CPS transformation for delimited continuations 103
8.9 Further reading 104

© in this web service Cambridge University Press & Assessment www.cambridge.org

CAMBRIDGE
Cambridge University Press & Assessment
978-1-009-76751-4 — Control Structures for Programming Languages
Xavier Leroy
Table of Contents
More Information

Part III From Exceptions to Algebraic Effects and Handlers 105

9 Exceptions 107

9.1 Programming with exceptions 107
9.2 Semantics of exceptions 110
9.3 ERS and CPS transformations for exceptions 111
9.4 Checked exceptions 113
9.5 Exception capabilities 117
9.6 Further reading 120

10 Effect Handlers for User-Defined Effects 121

10.1 From exceptions to effects 121
10.2 Effect handlers and delimited continuations 124
10.3 Control inversion 125
10.4 Cooperative multithreading 128
10.5 Semantics of effect handlers 131
10.6 CPS transformation for effect handlers 132
10.7 Further reading 134

11 Monads 135

11.1 Commonalities in semantics and program transformations 135
11.2 Monads and the computational lambda-calculus 137
11.3 Examples of monads 140
11.4 The monadic transformation 144
11.5 Programming with monads 146
11.6 Further reading 148

12 Algebraic Effects 150

12.1 Generic semantics for monadic programs 150
12.2 Free monads 153
12.3 Algebra reminder 156
12.4 Monads as algebraic structures 158
12.5 A language of algebraic effects 159
12.6 Semantic effect handlers 163
12.7 A language of algebraic effects and handlers 164
12.8 Further reading 166

Part IV Reasoning about Control and Effects 167

13 Type and Effect Systems 169

13.1 Type systems reminder 169
13.2 Simple types for advanced control structures 171
13.3 Issues with parametric polymorphism 174
13.4 A simple type and effect system for checked exceptions 176

© in this web service Cambridge University Press & Assessment www.cambridge.org

CAMBRIDGE
Cambridge University Press & Assessment
978-1-009-76751-4 — Control Structures for Programming Languages
Xavier Leroy
Table of Contents
More Information

13.5 Subtyping over effect types 179
13.6 Row polymorphism for effect types 180
13.7 A polymorphic type and effect system for algebraic effects 182
13.8 Limitations and extensions of row polymorphism 185
13.9 Further reading 187

14 Hoare Logic for Control Structures 188
14.1 Deductive verification reminder 188
14.2 Hoare logic: a program logic for structured control 189
14.3 Hoare logic for jumps 193
14.4 Hoare logic for coroutines and cooperative threads 197
14.5 Further reading 200

15 Separation Logic for Control Operators 201
15.1 Separation logic reminder 201
15.2 A separation logic for a functional-imperative language 202
15.3 A separation logic for callcc 205
15.4 A separation logic for linear undelimited continuations 207
15.5 A separation logic for effect handlers with linear continuations 211
15.6 Further reading 214
Bibliography 216
Index 226

© in this web service Cambridge University Press & Assessment www.cambridge.org