References

2. A. Alexandrescu, Modern C++ Design: Generic Programming and Design Patterns Applied (Addison-Wesley, Boston, Massachusetts, 2001), ISBN 978-0201704310
18. B. Ganter, Two basic algorithms in concept analysis, FB4-Preprint No. 831 (Technische Hochschule Darmstadt, June 1984)
32. C. Morgan, Programming from specifications (1998), http://web2.comlab.ox.ac.uk/oucl/publications/books/PfS/
43. B.W. Watson, Minimizing acyclic deterministic finite automata, Ph.D., FASTAR Research Group, Department of Computer Science, University of Pretoria, South Africa, 2011
Index

:=, see Assignment command
;., see Composition command
X-extension of a lattice, 204
w-path from a state, 234
abort, 22, 23, 31–33
skip, see Empty command
do . . . od, see Repetition command
if . . . fi, see Selection command

Acyclic deterministic Finite Automaton, 230
ADFA, see Acyclic deterministic Finite Automaton
Array, 49

Class
Abstraction, 46
Enrichment
Definition, 47
Generalisation, 46
Inheritance, 46
Refinement
Definition, 47
Specialisation, 46
Closure system, 201
Command, 17
Assignment, 24
Multiple, 25
Single, 24
Composition, 27
Definition, 27
Empty, 18
Definition, 18
Repetition, 32
Satisfies specification, 34
Selection, 30
Definition, 32

Confluence state, 233
Confluence-free set of states, 233
Correctness by construction, 6, 7, 46, 55, 117, 197, 198, 215, 219, 227, 232, 242, 246, 257
Cover relationship, 200

DbC, see Design by Contract
Depth level, 235
Design by Contract, 48
Deterministic finite automaton, 229
DFA, see Deterministic finite automaton

Examples
Binary search, 66
Dutch National Flag, 95
Exponentiation, 77
Integer logarithm approximation, 84
Linear search, 56
Longest segment, 103
MADFA Construction
Abstract algorithm, 237
Confluence-based, 250
Partial order based, 255
Trie-based, 244
Majority voting, 136
Maximal axial lines, 144
Maximal element, 60
Palindromes, 112
Pattern matching, 74
Raster circle, 127
Raster line, 117
Recursive
Search a list, 181
Evaluate an expression tree, 185
Find a factorial value, 177
Merge-sort, 191
SICL Construction, 205

GCL, see Guarded command language
Greatest lower bound, see Infimum
Guarded command language, 17

Additional commands
  as . . . sa, 50
  func . . . cnuf, 51
  proc . . . corp, 51
  Block notation, 50

Commands, see Command

Hoare triple, 14
  Extreme cases, 18

Infimum, 200

Definition of
  Loop, 33

Join, see Supremum

Lattice, 200
Least upper bound, see Supremum
Lexicographically greatest word, 236
Lower bound, 200

MADFA, see Minimum acyclic deterministic finite automata
Meet, see Infimum
Minimum acyclic deterministic finite automata, 231
Minimum path length, 235

Pairwise inequivalent
  Set of states, 236
  States, 236
Partial order, 198
Postcondition, 10, 14–16, 18, 23, 24, 26, 31, 32, 36, 39, 44, 45, 48, 49
Precondition, 10, 14–16, 18, 25, 36–38, 47, 48
  Weakest, 16, 20, 22, 23, 25–29, 31–33, 37
  Extreme cases, 21
Predicate

Statesₚ, 11
  As set of states, 10
  Strength, 12
  Strong, 11
  Weak, 11
Predicate calculus, 7, 8, 10
Propositional calculus, 8

Recursion, 161, 162, 173–175, 185, 194, 195
  Base case, 174
  Deriving recursive programs, 173
  Variant, 173
Refinement
  And weakest precondition, 37
  Class, see Class refinement
  Definition, 35
  Morgan’s refinement calculus, 48
Rules, 34
  Assignment, 38
  Composition, 39
  Following assignment, 41
  Parameterless call, 162
  Pass-by-result call, 167
  Pass-by-value call, 164
  Pass-by-value-result call, 168
  Procedures, 45
  Selection, 44
  Skip, 36
  Strengthen postcondition, 35
  Weaken precondition, 36

Right language of a state, 234

Sequence, 49
Set intersection-closed lattice, 201
Shortest word length of an ADFA, 235
SICL, see Set intersection-closed lattice
Specification, 14
  Command satisfies, 34
State equivalence, 236
Strategy for developing a loop, 44
Substitution
  General, 25
  Single, 12
Successor states, 235
Supremum, 199

Trie, 233
Upper bound, 199
Variant, 42