

Bibliography

- Harold Abelson and Gerald Jay Sussman. 1985. *Structure and Interpretation of Computer Programs*. McGraw-Hill, New York.
- Alfred V. Aho, Brian W. Kernighan, and Peter J. Weinberger. 1988. *The AWK Programming Language*. Addison-Wesley, Reading, MA.
- Alfred V. Aho, Ravi Sethi, and Jeffrey D. Ullman. 1986. *Compilers: Principles, Techniques, and Tools*. Addison-Wesley.
- Alfred V. Aho and Jeffrey D. Ullman. 1972. *The theory of parsing, translation, and compiling*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA. ISBN 0-13-914556-7.
- Andrew W. Appel. 1987 (June). Garbage collection can be faster than stack allocation. *Information Processing Letters*, 25(4):275–279.
- Andrew W. Appel. 1989 (February). Simple generational garbage collection and fast allocation. *Software—Practice & Experience*, 19(2):171–183.
- Andrew W. Appel. 1998. *Modern Compiler Implementation*. Cambridge University Press, Cambridge, UK. Available in three editions: C, Java, and ML.
- Henry G. Baker. 1978 (April). List processing in real-time on a serial computer. *Communications of the ACM*, 21(4):280–294.
- J. G. P. Barnes. 1991. *Programming in Ada (third edition)*. Addison-Wesley, Wokingham, England.
- W. Barrett, R. Bates, D. Gustafson, and J. Couch. 1986. *Compiler Construction: Theory and Practice*. SRA, Chicago, 2nd edition.
- Joel F. Bartlett. 1988 (February). Compacting garbage collection with ambiguous roots. Technical Report 88/2, DEC WRL, 100 Hamilton Avenue, Palo Alto, California 94301.
- Kent Beck. 1997. *Smalltalk Best Practice Patterns*. Prentice Hall.
- Richard Bird and Philip Wadler. 1988. *Introduction to Functional Programming*. Prentice Hall, New York.
- G. M. Birtwistle, O.-J. Dahl, B. Myhrhaug, and K. Nygaard. 1973. *Simula Begin*. Van Nostrand-Reinhold, New York, 1 edition.
- Judy M. Bishop. 1986. *Data Abstraction in Programming Languages*. Addison-Wesley. ISBN 0-201-14222-8.

- D. G. Bobrow, L. G. DeMichiel, R. P. Gabriel, S. E. Keene, G. Kiczales, and D. A. Moon. 1988 (September). Common lisp object system specification. *ACM SIGPLAN Notices*, 23(SI). ISSN 0362-1340.
- Daniel G. Bobrow, Ken Kahn, Gregor Kiczales, Larry Masinter, M. Stefik, and F. Zdybel. 1986 (November). Common Loops, merging Lisp and object-oriented programming. *ACM SIGPLAN Notices*, 21(11):17–29. ISSN 0362-1340.
- Hans-Juergen Boehm and Mark Weiser. 1988 (September). Garbage collection in an uncooperative environment. *Software—Practice & Experience*, 18(9):807–820. Hans Boehm continues to maintain this collector, and a link can usually be found on his web pages, e.g., at http://www.hpl.hp.com/personal/Hans_Boehm/gc as of January 2003.
- Per Brinch Hansen. 1994 (June). Multiple-length division revisited: A tour of the minefield. *Software—Practice & Experience*, 24(6):579–601.
- Timothy Budd. 1987. *A Little Smalltalk*. Addison-Wesley, Reading, MA. ISBN 0-201-10698-1.
- W. H. Burge. 1975. *Recursive Programming Techniques*. Addison-Wesley, Reading. ISBN 0-201-14450-6.
- R. M. Burstall, J. S. Collins, and R. J. Popplestone. 1971. *Programming in POP-2*. Edinburgh University Press, Edinburgh, UK.
- Rod M. Burstall, David B. MacQueen, and Donald T. Sannella. 1980 (August). Hope: An experimental applicative language. In *Conference Record of the 1980 LISP Conference*, pages 136–143. ACM, ACM.
- Lawrence Byrd. 1980. Understanding the control flow of Prolog programs. In S.-A. Tarnlund, editor, *Proceedings of the Logic Programming Workshop*, pages 127–138. See also University of Edinburgh technical report 151.
- Byte. 1981 (August). Special issue on Smalltalk. *Byte Magazine*, 6(8). ISSN 0360-5280.
- Luca Cardelli. 1987 (April). Basic polymorphic typechecking. *Science of Computer Programming*, 8(2):147–172. ISSN 0167-6423. URL <http://research.microsoft.com/Users/luca/Papers/BasicTypechecking.ps>.
- Luca Cardelli. 1997. Type systems. In Jr. Allen B. Tucker, editor, *The Computer Science and Engineering Handbook*, chapter 103, pages 2208–2236. CRC Press, Boca Raton, FL.
- Chin-Liang Chang and Richard Char-Tung Lee. 1973. *Symbolic Logic and Mechanical Theorem Proving*. Academic Press, New York.
- K. L. Clark and S.-A. (Eds.) Tarnlund. 1983. *Logic Programming*. Academic Press.
- William D. Clinger. 1998 (May). Proper tail recursion and space efficiency. *Proceedings of the ACM SIGPLAN '98 Conference on Programming Language Design and Implementation*, in *SIGPLAN Notices*, 33(5):174–185. ISSN 0362-1340. URL <http://www.acm.org:80/pubs/citations/proceedings/pldi/277650/p174-clinger/>.
- W. F. Clocksin and C. S. Mellish. 1994. *Programming in Prolog*. Springer-Verlag, Berlin, 4th edition. ISBN 3-540-58350-5.

- Jacques Cohen. 1981 (September). Garbage collection of linked data structures. *Computing Surveys*, 13(3):341–367.
- Jacques Cohen. 1988 (January). A view of the origins and development of Prolog. *Communications of the ACM*, 31(1):26–36. ISSN 0001-0782. URL <http://www.acm.org/pubs/toc/Abstracts/0001-0782/35045.html>.
- Norman H. Cohen. 1986. *Ada as a Second Language*. MacGraw-Hill (New York NY).
- A. Colmerauer, H. Kanoui, R. Pasero, and P. Roussel. 1973 (November). Un systeme de communication homme-machine en francais. Technical report, Groupe d'Intelligence Artificielle, Université d'Aix-Marseille II.
- Thomas J. Conroy and Eduardo Pelegri-Llopart. 1982. An assessment of method-lookup caches for Smalltalk-80 implementations. In *Smalltalk-80: Bits of History, Words of Advice*, chapter 13, pages 239–247. Addison-Wesley, Reading, MA.
- Brad J. Cox. 1986. *Object-Oriented Programming - an Evolutionary Approach*. Addison-Wesley, Reading, 1 edition. ISBN 0-201-10393-1.
- Ole-Johan Dahl and C. A. R. Hoare. 1972. Hierarchical program structures. In *Structured Programming*, chapter 3, pages 175–220. Academic Press.
- Luis Damas and Robin Milner. 1982. Principal type-schemes for functional programs. In *Conference Record of the 9th Annual ACM Symposium on Principles of Programming Languages*, pages 207–12, New York. ACM Press.
- Doug DeGroot and Gary Lindstrom, editors. 1986. *Logic Programming: Relations, Functions, and Equations*. Prentice-Hall.
- Peter Deutsch and Alan M. Schiffman. 1984 (January). Efficient implementation of the Smalltalk-80 system. In *Conference Record of the 11th Annual ACM Symposium on Principles of Programming Languages*, pages 297–302. ACM, ACM.
- R. Kent Dybvig. 1987. *The SCHEME Programming Language*. Prentice-Hall, Upper Saddle River, NJ 07458, USA. ISBN 0-13-791864-X (paperback).
- Matthias Felleisen, Robert Bruce Findler, Matthew Flatt, and Shriram Krishnamurthi. 2001. *How to Design Programs: An Introduction to Programming and Computing*. MIT Press, Cambridge, MA.
- Matthias Felleisen, Robert Bruce Findler, Matthew Flatt, and Shriram Krishnamurthi. 2004. The structure and interpretation of the Computer Science curriculum. *Journal of Functional Programming*, 14(4):365–378.
- Robert R. Fenichel and Jerome C. Yochelson. 1969. A LISP garbage-collector for virtual-memory computer systems. *Communications of the ACM*, 12(11):611–12.
- David Flanagan and Yukihiro Matsumoto. 2008. *The Ruby Programming Language*. O'Reilly. ISBN 9780596516178.
- Daniel P. Friedman and Matthias Felleisen. 1996. *The Little Schemer*. MIT Press, fourth edition. ISBN 0-0262-56099-2.

- Daniel P. Friedman, Christopher T. Haynes, and Eugene E. Kohlbecker. 1984. Programming with continuations. In P. Pepper, editor, *Program Transformation and Programming Environments*, pages 263–274. Springer-Verlag.
- Kokichi Futatsugi, Joseph A. Goguen, Jean-Pierre Jouannaud, and Jose Meseguer. 1985 (January). Principles of OBJ2. In *Conference Record of the Twelfth Annual ACM Symposium on Principles of Programming Languages*, pages 52–66. ACM, ACM.
- J. H. Gallier. 1986. *Logic for Computer Science: Foundations of Automatic Theorem Proving*, volume 5 of *Computer Science and Technology Series*. Harper & Row, New York.
- Emden Gansner and John Reppy, editors. 2002. *The Standard ML Basis Library*. Cambridge University Press, New York, NY.
- Jeremy Gibbons and Geraint Jones. 1998 (September). The under-appreciated unfold. *Proceedings of the 1998 ACM SIGPLAN International Conference on Functional Programming*, in *SIGPLAN Notices*, 34(1):273–279.
- J. Goguen, J. Thatcher, and E. Wagner. 1979. An initial algebra approach to the specification, correctness, and implementation of abstract data types. In R. Yeh, editor, *Current Trends in Programming Methodology IV*, pages 80–149. Prentice-Hall.
- A. Goldberg. 1983. *Smalltalk 80: The Interactive Programming Environment*. Addison-Wesley.
- Adele Goldberg and David Robson. 1983. *Smalltalk-80: The Language and Its Implementation*. Addison-Wesley, Reading, MA.
- James Gosling, Bill Joy, and Guy Steele. 1997. *The Java Language Specification*. The Java Series. Addison-Wesley. ISBN 0-201-63451-1. URL <http://www.aw.com/cp/gosling-joy-et-al.html>.
- Paul Graham. 1993. *On Lisp: Advanced Techniques for Common Lisp*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA. ISBN 0130305529. May be downloadable from <http://www.paulgraham.com/onlisp.html>.
- David Gries. 1981. *The Science of Programming*. Springer-Verlag.
- Ralph E. Griswold and Madge T. Griswold. 1996. *The Icon Programming Language*. Peer-to-Peer Communications, San Jose, CA, third edition. ISBN 1-57398-001-3.
- Dirk Grunwald and Benjamin G. Zorn. 1993 (August). CustoMalloc: Efficient synthesized memory allocators. *Software—Practice & Experience*, 23(8):851–869. ISSN 0038-0644. URL http://www.cs.colorado.edu/homes/grunwald/public_html/SPE93-customalloc.ps.
- J. V. Guttag, E. Horwitz, and D. R. Musser. 1978 (December). Abstract data type and software validation. *Communications of the ACM*, 21(12):1048–1064.
- David R. Hanson. 1996. *C Interfaces and Implementations*. Addison Wesley.
- Brian Harvey and Matthew Wright. 1994. *Simply Scheme: Introducing Computer Science*. MIT Press, Cambridge, Massachusetts, U.S.A. ISBN 0-262-08226-8.

- Peter Henderson. 1980. *Functional Programming: Application and Implementation*. International Series in Computer Science. Prentice-Hall, Inc., Englewood Cliffs, N.J.
- J. Roger Hindley. 1969. The principal type scheme of an object in combinatory logic. *Transactions of the American Mathematical Society*, 146:29–60.
- C. A. R. Hoare. 1972. Proof of correctness of data representations. *Acta Informatica*, 1: 271–281.
- C. A. R. Hoare. 1989. *Essays in Computing Science*. Prentice Hall.
- C. A. R. Hoare, I. J. Hayes, He Jifeng, C. C. Morgan, A. W. Roscoe, J. W. Sanders, I. H. Sørensen, J. M. Spivey, and B. A. Sufrin. 1987 (August). Laws of programming. *Communications of the ACM*, 30(8):672–686. ISSN 0001-0782. URL <http://www.acm.org/pubs/toc/Abstracts/0001-0782/27653.html>. See corrigendum in the September 1987 CACM.
- C. J. Hogger. 1984. *Introduction to Logic Programming*. Academic Press, London. ISBN 0-12-352090-8.
- John E. Hopcroft and Jeffrey D. Ullman. 1979. *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley, Reading, MA.
- John Hughes. 1995. The design of a pretty-printing library. In Johan Jeuring and Erik Meijer, editors, *Advanced Functional Programming, LNCS volume 925*, pages 53–96. Springer Verlag.
- Roberto Ierusalimsky. 2003 (December). *Programming in Lua*. Lua.org. ISBN 85-903798-1-7.
- Dan Ingalls, Ted Kaehler, John Maloney, Scott Wallace, and Alan Kay. 1997 (October 5–9). Back to the future: The story of Squeak - a practical Smalltalk written in itself. In *Proceedings of the ACM SIGPLAN Conference on Object-Oriented Programming Systems, Languages and Applications (OOPSLA-97)*, volume 32, 10 of *ACM SIGPLAN Notices*, pages 318–326, New York. ACM Press.
- Ada. 1995 (January). *Ada 95 Reference Manual. The Language. The Standard Libraries*. International Organization for Standardization. URL <http://www.adahome.com/Resources/refs/rm95.html>. ANSI/ISO/IEC-8652:1995.
- Ralph E. Johnson, Justin O. Graver, and Lawrence W. Zurawski. 1988. TS: An optimizing compiler for Smalltalk. In Norman Meyrowitz, editor, *OOPSLA'88: Object-Oriented Programming Systems, Languages and Applications: Conference Proceedings*, pages 18–26. ISBN 0-89791-284-5.
- Mark S. Johnstone and Paul R. Wilson. 1998 (October). The memory fragmentation problem: Solved? *Proceedings of the First International Symposium on Memory Management*, in *SIGPLAN Notices*, 34(3):26–36.
- Cliff B. Jones. 1980. *Software Development A Rigorous Approach*. Prentice-Hall International, Englewood Cliffs, New Jersey.
- Mark P. Jones. 1999 (October). Typing Haskell in Haskell. In *Proceedings of the 1999 Haskell Workshop*, Paris, France. Published in Technical Report UU-CS-1999-28, Department of Computer Science, University of Utrecht. Additional resources at <http://www.cse.ogi.edu/~mpj/thih>.

- Richard Jones and Rafael Lins. 1996. *Garbage collection: algorithms for automatic dynamic memory management*. Wiley, New York, NY, USA. ISBN 0-471-94148-4. URL http://www.ukc.ac.uk/computer_science/Html/Jones/gc.html. Reprinted in 1999 with improved index and corrected errata.
- Ted Kaehler and Dave Patterson. 1986. *A Taste of Smalltalk*. W. W. Norton and Company, New York.
- S. Kamin. 1983 (January). Final data types and their specification. *ACM Transactions on Programming Languages and Systems*, 5(1):97–123.
- A. C. Kay. 1993 (March). The early history of Smalltalk. *SIGPLAN Notices*, 28(3):69–95. ISSN 0362-1340.
- Richard Kelsey, William Clinger, and Jonathan Rees. 1998 (September). Revised⁵ report on the algorithmic language Scheme. *SIGPLAN Notices*, 33(9):26–76. ISSN 0362-1340.
- Brian W. Kernighan and Dennis M. Ritchie. 1988. *The C Programming Language*. Prentice Hall, Englewood Cliffs, NJ, second edition.
- Donald E. Knuth. 1965 (December). On the translation of languages from left to right. *Information and Control*, 8(6):607–639.
- Donald E. Knuth. 1973. *Fundamental Algorithms*, volume 1 of *The Art of Computer Programming*. Addison-Wesley, Reading, MA, second edition.
- Donald E. Knuth. 1974 (December). Structured programming with **go to** statements. *ACM Computing Surveys*, 6(4):261–301.
- Donald E. Knuth. 1981. *Seminumerical Algorithms*, volume 2 of *The Art of Computer Programming*. Addison-Wesley, Reading, MA, second edition.
- Donald E. Knuth. 1984. Literate programming. *The Computer Journal*, 27(2):97–111.
- Andrew Koenig. 1994. An anecdote about ML type inference. In *Proceedings of the USENIX 1994 Very High Level Languages Symposium, VHLLS'94*, pages 1–1, Berkeley, CA, USA. USENIX Association.
- Robert A. Kowalski. 1974. Predicate logic as a programming language. *Proc. IFIP 4*, pages 569–574.
- Robert A. Kowalski. 1979 (July)a. Algorithm = logic + control. *Communications of the ACM*, 22(7):424–436. ISSN 0001-0782.
- Robert A. Kowalski. 1979b. *Logic for Problem Solving*. The Computer Science Library, Artificial Intelligence Series. North Holland, New York.
- Robert A. Kowalski. 1988 (January). The early years of logic programming. *Communications of the ACM, CACM*, 31(1):38–43.
- Glenn Krasner, editor. 1983. *Smalltalk-80: Bits of History, Words of Advice*. Addison-Wesley Longman, Boston, Mass. ISBN 0-201-11669-3.
- P. J. Landin. 1966. The next 700 programming languages. *Communications of the ACM*, 9(3):157–66.

- Henry Lieberman and Carl Hewitt. 1983 (June). A real-time garbage collector based on the lifetimes of objects. *Communications of the ACM*, 26(6):419–429.
- B. H. Liskov and S. N. Zilles. 1975 (March). Specification techniques for data abstractions. *IEEE Transactions on Software Engineering*, 1(1):7–19.
- Barbara Liskov, Russell Atkinson, Toby Blook, Eliot Moss, J. Craig Schaffert, Robert Scheifler, and Alan Snyder. 1981. *CLU reference manual*, volume 114 of *Lecture Notes in Computer Science*. Springer-Verlag, Berlin, Germany.
- Barbara Liskov and John Guttag. 1986. *Abstraction and Specification in Program Development*. MIT Press / McGraw-Hill.
- Barbara Liskov, Alan Snyder, Russell Atkinson, and Craig Schaffert. 1977 (August). Abstraction mechanisms in CLU. *Communications of the ACM*, 20(8):564–576. ISSN 0001-0782. Also published in/as: In “Readings in Object-Oriented Database Systems” edited by S.Zdonik and D.Maier, Morgan Kaufman, 1990.
- Barbara H. Liskov and Alan Snyder. 1979 (November). Exception handling in CLU. *IEEE Transactions on Software Engineering*, SE-5(6):546–558.
- John Wylie Lloyd. 1984. *Foundations of Logic Programming*. Springer-Verlag, New York.
- Jacques Loeckx, Kurt Sieber, and Ryan D. Stansifer. 1987. *The Foundations of Program Verification*. Wiley-Teubner Series in Computer Science. Wiley, Chichester, second edition edition. ISBN 0-471-91282-4 (Wiley) 3-519-121018 (Teubner).
- David Maier and David S. Warren. 1988. *Computing with Logic*. Benjamin/Cummings, Menlo Park.
- Zohar Manna. 1974. *Mathematical Theory of Computation*. McGraw-Hill, New York. ISBN 0-07-039910-7.
- Conor McBride and Ross Paterson. 2008 (January). Applicative programming with effects. *Journal of Functional Programming*, 18(1):1–13.
- John McCarthy. 1960 (April). Recursive functions of symbolic expressions and their computation by machine, part I. *Communications of the ACM*, 3(4):184–195.
- John McCarthy. 1962. *Lisp 1.5 Programmer's Manual*. Cambridge, Mass.
- A. D. McGettrick. 1982. *Program Verification using Ada*. Number 13 in Cambridge Computer Science Texts. Cambridge University Press, Cambridge, UK.
- Bertrand Meyer. 1988. *Object-Oriented Software Construction*. Prentice Hall International, London. Second edition published as (Meyer 1997).
- Bertrand Meyer. 1992. *Eiffel: The Language*. Prentice Hall International, London.
- Bertrand Meyer. 1997. *Object-Oriented Software Construction*. Prentice-Hall, Englewood Cliffs, NJ, second edition. ISBN 0-13-629155-4. URL http://www.prenhall.com/allbooks/ptr_0136291554.html.
- Robin Milner. 1978 (December). A theory of type polymorphism in programming. *Journal of Computer and System Sciences*, 17:348–375.

- Robin Milner. 1983. How ML evolved. *Polymorphism—The ML/LCF/Hope Newsletter*, 1 (1).
- Marvin L. Minsky. 1963 (December). A lisp garbage collector algorithm using serial secondary storage. Technical Report Memo 58, Project MAC, MIT, Cambridge, MA.
- David A. Moon. 1986 (October). Object-oriented programming with flavours. In *Proceedings of the ACM Conference on Object-Oriented Programming Systems, Languages, and Applications*, pages 1–8.
- George C. Necula, Scott McPeak, and Westley Weimer. 2002 (January). CCured: Type-safe retrofitting of legacy code. *Conference Record of the 29th Annual ACM Symposium on Principles of Programming Languages*, in *SIGPLAN Notices*, 37(1):128–139.
- Greg Nelson, editor. 1991. *Systems Programming with Modula-3*. Prentice Hall, Englewood Cliffs, NJ.
- Nils J. Nilsson. 1980. *Principles of artificial intelligence*. Tioga/Morgan Kaufman, 476 pages; ISBN 0-934613-10-9; US&Canada \$54.95 CALL NUMBER: Q335 .N515.
- Martin Odersky, Martin Sulzmann, and Martin Wehr. 1999. Type inference with constrained types. *Theory and Practice of Object Systems*, 5(1):35–55.
- Melissa E. O’Neill. 2009 (January). The genuine sieve of Eratosthenes. *Journal of Functional Programming*, 19(1):95–106. ISSN 0956-7968.
- Derek C. Oppen. 1980 (October). Prettyprinting. *ACM Transactions on Programming Languages and Systems*, 2(4):465–483.
- David L. Parnas. 1972 (May)a. A technique for software module specification with examples. *Communications of the ACM*, 15(5):330–336.
- David Lorge Parnas. 1972 (December)b. On the criteria for decomposing systems into modules. *Communications of the ACM*, 15(12):1053–1058.
- J. Payne. 1982. *Introduction to Simulation: Programming Techniques and Methods of Analysis*. McGraw-Hill, New York.
- Simon Peyton Jones, Dimitrios Vytiniotis, Stephanie Weirich, and Mark Shields. 2007. Practical type inference for arbitrary-rank types. *Journal of Functional Programming*, 17 (1):1–82.
- Benjamin C. Pierce. 2002. *Types and Programming Languages*. MIT Press, Cambridge, MA.
- Rob Pike. 1990 (July). The implementation of newsqueak. *Software—Practice & Experience*, 20(7):649–659. URL <http://plan9.bell-labs.com/cm/cs/doc/88/1-a.ps.gz>.
- Gordon D. Plotkin. 1981 (September). A structural approach to operational semantics. Technical Report DAIMI FN-19, Department of Computer Science, Aarhus University, Aarhus, Denmark.
- François Pottier and Didier Rémy. 2005. *The Essence of ML Type Inference*, chapter 10, pages 389–489. MIT Press.

- Todd A. Proebsting. 1997 (June). Simple translation of goal-directed evaluation. *Proceedings of the ACM SIGPLAN '97 Conference on Programming Language Design and Implementation*, in *SIGPLAN Notices*, 32(5):1-6.
- Norman Ramsey. 1994 (September). Literate programming simplified. *IEEE Software*, 11(5):97-105.
- Norman Ramsey. 1999. Eliminating spurious error messages using exceptions, polymorphism, and higher-order functions. *Computer Journal*, 42(5):360-372.
- E. Reingold and R. Reingold. 1988. *Pascal Algorithms: An Introduction to Programming*. Scott, Foresman, Illinois.
- John Reynolds. 1972 (August). Definitional interpreters for higher-order programming languages. In *Proceedings of the 25th ACM National Conference*, pages 717-740. ACM.
- John C. Reynolds. 1981. *The Craft of Programming*. Prentice Hall, Englewood Cliffs. ISBN 0-13-188862-5.
- John C. Reynolds. 1998 (December). Definitional interpreters for higher-order programming languages. *Higher-Order and Symbolic Computation*, 11(4):363-397. Reprinted from the proceedings of the 25th ACM National Conference (1972).
- G. A. Ringwood. 1988 (January). Parlog86 and the dining logicians. *Communications of the ACM*, 31(1):10-25. ISSN 0001-0782. URL <http://www.acm.org/pubs/toc/Abstracts/0001-0782/35044.html>.
- E. S. Roberts. 1986. *Thinking Recursively*. John Wiley & Sons, New York.
- J. A. Robinson. 1965 (January). A machine-oriented logic based on resolution principle. *Journal of the ACM*, 12(1):23-49.
- J. A. Robinson. 1983. Logic programming - past, present and future. *New Generation Comput. (Japan)* ISSN: 0288-3635, 1(2):107-24. QA 76 N 48.
- J. S. Rohl. 1984. *Recursion via Pascal*, volume 19 of *Cambridge Computer Science Texts*, Editors: E. S. Page and C. M. Reeves and D. E. Conway. Cambridge University Press, Cambridge, UK. ISBN 0-521-26329-8 (hardcover), 0-521-26934-2 (paperback).
- Neil C. Rowe. 1988. *Artificial Intelligence through Prolog*. Prentice-Hall publishers, .; ACM CR 8903-0118.
- Colin Runciman and David Wakeling. 1993 (April). Heap profiling of lazy functional programs. *Journal of Functional Programming*, 3(2):217-246.
- C. Schaffert, T. Cooper, B. Bullis, M. Kilian, and C. Wilpolt. 1986 (November). An introduction to trellis/owl. *OOPSLA'86 Proceedings*, 21(11):9-16.
- Kurt J. Schmucker. 1986. *Object-Oriented Programming for the Macintosh*. Hayden Book Company.
- Herbert Schorr and William M. Waite. 1967 (August). An efficient machine-independent procedure for garbage collection in various list structures. *Communications of the ACM*, 10(8):501-506.
- T. Schriber. 1974. *Discrete System Simulation*. Wiley, New York.

- Robert Sedgewick. 1988. *Algorithms*. Addison-Wesley, second edition.
- Manuel Serrano and Hans-Juergen Boehm. 2000 (September). Understanding memory allocation of Scheme programs. *Proceedings of the ACM SIGPLAN International Conference on Functional Programming (ICFP'00)*, in *SIGPLAN Notices*, 35(9): 245–256. ISSN 0362-1340. URL <http://www.acm.org/pubs/citations/proceedings/fp/351240/p245-serrano/>; <http://www.acm.org/pubs/articles/proceedings/fp/351240/p245-serrano/p245-serrano.pdf>.
- Tim Sheard. 2001 (September). Generic unification via two-level types and parameterized modules. In *Proceedings of the Sixth ACM SIGPLAN International Conference on Functional Programming (ICFP'01)*, pages 86–97.
- J. Siekmann and G. Wrightson. 1983. *Automation of Reasoning I and II*. Springer-Verlag.
- Stephen Slade. 1987. *The T programming Language*. Prentice-Hall Inc., Englewood Cliffs, N.J.
- Frederick Smith and Greg Morrisett. 1999 (March). Comparing mostly-copying and mark-sweep conservative collection. *Proceedings of the First International Symposium on Memory Management (ISMM'98)*, in *SIGPLAN Notices*, 34(3):68–78.
- Michael Sperber, R. Kent Dybvig, Matthew Flatt, Anton van Straaten, Robby Findler, and Jacob Matthews. 2009. Revised⁶ Report on the algorithmic language Scheme. *Journal of Functional Programming*, 19(Supplement S1):1–301. URL <http://dx.doi.org/10.1017/S0956796809990074>.
- T. A. Standish. 1980. *Data Structure Techniques*. Addison-Wesley, Reading, MA.
- Guy Lewis Steele, Jr. 1984. *Common Lisp: The Language*. Digital Press.
- Guy Lewis Steele, Jr and Gerald Jay Sussman. 1976 (March). Lambda: The ultimate imperative. Technical Report AIM-353, Massachusetts Institute of Technology. URL <ftp://publications.ai.mit.edu/ai-publications/0-499/AIM-353.tiff.tar.gz>.
- Guy Lewis Steele, Jr and Gerald Jay Sussman. 1978 (May). The art of the interpreter or, the modularity complex (parts zero, one, and two). Technical Report AIM-453, Massachusetts Institute of Technology. URL <ftp://publications.ai.mit.edu/ai-publications/0-499/AIM-453.ps>.
- L. Sterling and E. Shapiro. 1986. *The Art of Prolog*. MIT Press, Cambridge, Mass. ISBN 0-262-19250-0.
- Bjarne Stroustrup. 1997. *The C++ Programming Language*. Addison-Wesley, Reading, MA, third edition. ISBN 0-201-88954-4.
- Gerald Jay Sussman and Guy Lewis Steele, Jr. 1975 (December). Scheme: An interpreter for extended lambda calculus. Technical Report 349, MIT. Reprinted in (Sussman and Steele 1998).
- Gerald Jay Sussman and Guy Lewis Steele, Jr. 1998 (December). Scheme: An interpreter for extended lambda calculus. *Higher-Order and Symbolic Computation*, 11(4):405–439. ISSN 1388-3690. URL <http://www.wkap.nl/oasis.htm/193825>. Originally Technical Report 349, MIT, December 1975.

- D. S. Touretzky. 1984. *LISP: A Gentle Introduction to Symbolic Computation*. Harper & Row, Pubs., New York, NY. ISBN 0-06-046657-X.
- David Ungar. 1984 (May). Generation scavenging: A non-disruptive high performance storage reclamation algorithm. *Proceedings of the ACM SIGSOFT/SIGPLAN Symposium on Practical Software Development Environments*, in *SIGPLAN Notices*, 19(5):157–167.
- US DoD. 1983 (February). *American National Standard reference manual for the Ada programming language: ANSI/MIL-STD-1815A-1983*. United States. Dept. of Defense and American National Standards Institute, 1430 Broadway, New York, NY 10018, USA.
- Guido van Rossum. 1998. A tour of the Python language. In R. Ege, M. Singh, and B. Meyer, editors, *Proceedings. Technology of Object-Oriented Languages and Systems, TOOLS-23*, page 370, 1109 Spring Street, Suite 300, Silver Spring, MD 20910, USA. IEEE Computer Society Press. ISBN 0-8186-8383-X.
- Dimitrios Vytiniotis, Simon Peyton Jones, and Tom Schrijvers. 2010. Let should not be generalised. In *Proceedings of the 5th ACM SIGPLAN workshop on Types in language design and implementation, TLDI '10*, pages 39–50. ACM.
- Philip Wadler. 1999. A prettier printer. Unpublished note available from the author's Web site.
- Daniel C. Wang, Andrew W. Appel, Jeff L. Korn, and Christopher S. Serra. 1997 (October). The Zephyr Abstract Syntax Description Language. In *Proceedings of the 2nd USENIX Conference on Domain-Specific Languages*, pages 213–227, Santa Barbara, CA.
- R. L. Wexelblat, editor. 1981. *History of Programming Languages*. Academic Press, New York. ISBN 0-12-745040-8.
- Robert Wilensky. 1986. *Common LISPcraft*. W. W. Norton & Co., New York, NY, USA. ISBN 0-393-95544-3.
- Paul R. Wilson. 1992 (September). Uniprocessor garbage collection techniques. In *Proceedings of the International Workshop on Memory Management*, pages 1–42, St. Malo, France. Published as Lecture Notes in Computer Science 637, Springer-Verlag.
- Paul R. Wilson, Mark S. Johnstone, Michael Neely, and David Boles. 1995 (September). Dynamic storage allocation: A survey and critical review. In Henry Baker, editor, *Proceedings of International Workshop on Memory Management, LNCS volume 986*, Kinross, Scotland. Springer-Verlag. URL <ftp://ftp.cs.utexas.edu/pub/garbage/allocsrv.ps>.
- Terry Winograd. 1972. *Understanding Natural Language*. Academic Press, New York.
- P. H. Winston and B. K. P. Horn. 1984. *Lisp Second Edition*. Addison-Wesley, Reading. ISBN 0-201-08372-8.
- Patrick H. Winston. 1977. *Artificial Intelligence*. Addison Wesley Publishing Co., Reading, Mass.
- Niklaus Wirth. 1971 (April). Program development by stepwise refinement. *Communications of the ACM*, 14(4):221–227. ISSN 0001-0782. URL <http://www.acm.org/classics/dec95/>.
- Niklaus Wirth. 1977 (November)a. What can we do about the unnecessary diversity of notation for syntactic definitions? *Communications of the ACM*, 20(11):822–823.

- Niklaus Wirth. 1977 (November)b. What can we do about the unnecessary diversity of notation for syntactic definitions? *Communications of the ACM*, 20(11):822–823. [This entry is the first match for query `wirth:unnecessary-diversity`, which produced 2 matches.]
- Niklaus Wirth. 1982. *Programming in Modula-2*. Springer, Berlin. ISBN 0-387-11674-5.
- Larry Wos, Ross Overbeek, Ewing Lusk, and Jim Boyle. 1984. *Automated Reasoning—Introduction and Applications*. Prentice-Hall, Englewood Cliffs.
- William A. Wulf, R. L. London, and Mary Shaw. 1976 (December). An introduction to the construction and verification of alphas programs. *IEEE Transactions on Software Engineering*, 2(4):253–265.
- Hongwei Xi and Frank Pfenning. 1998 (May). Eliminating array bound checking through dependent types. *Proceedings of the ACM SIGPLAN '98 Conference on Programming Language Design and Implementation*, in *SIGPLAN Notices*, 33(5):249–257. ISSN 0362-1340. URL <http://www.acm.org:80/pubs/citations/proceedings/pldi/277650/p249-xi/>.
- Benjamin Zorn. 1993 (July). The measured cost of conservative garbage collection. *Software—Practice & Experience*, 23(7):733–756.