

BIBLIOGRAPHY

- [1] Harold Abelson, Gerald Jay Sussman, and Julie Sussman. *Structure and Interpretation of Computer Programs*. McGraw-Hill Book Company, 1985.
- [2] Alan Baddeley. Working memory. *Science*, 255:556–9, 31 January 1992.
- [3] Manfred Broy. Experiences with software specification and verification using LP, the Larch Proof Assistant. Technical Report 93, Digital Equipment Corporation Systems Research Center, 130 Lytton Avenue, Palo Alto, CA 94301, November 1992.
- [4] Stephen A. Cook. Soundness and completeness of an axiom system for program verification. *SIAM Journal on Computing*, 7(1):70–90, February 1978.
- [5] Laurence Peter Deutsch. *An Interactive Program Verifier*. PhD thesis, Department of Computer Science, University of California, Berkeley, May 1973.
- [6] Edsger Wybe Dijkstra. *A Discipline of Programming*. Prentice-Hall Series in Automatic Computation. Prentice-Hall, Englewood Cliffs, NJ, 1976.
- [7] Stephen H. Edwards and William F. Ogden. Personal communication, October 1995.
- [8] Herbert B. Enderton. *A Mathematical Introduction to Logic*. Academic Press, 1972.
- [9] George W. Ernst, Raymond J. Hookway, James A. Menegay, and William F. Ogden. Semantics of programming languages for modular verification. Technical Report CES-85-04, Computer Engineering and Science Department, Case Western Reserve University, 1985.
- [10] George W. Ernst, Raymond J. Hookway, James A. Menegay, and William F. Ogden. Modular verification of Ada generics. *Computer Languages*, 16(3/4):259–280, 1991.

- [11] George W. Ernst, Raymond J. Hookway, and William F. Ogden. Modular verification of data abstractions with shared realizations. *IEEE Transactions on Software Engineering*, 20(4):288–307, April 1994.
- [12] George W. Ernst, Jainendra K. Navlakha, and William F. Ogden. Verification of programs with Procedure-Type parameters. *Acta Informatica*, 18:149–169, 1982.
- [13] Robert W. Floyd. Assigning meanings to programs. In J. T. Schwartz, editor, *Proceedings of Symposia in Applied Mathematics*, volume 19, pages 19–32, Providence, Rhode Island, 1967. American Mathematical Society.
- [14] Stephen J. Garland and John V. Guttag. A guide to LP, the Larch Prover. Technical Report 82, Digital Equipment Corporation Systems Research Center, 130 Lytton Avenue, Palo Alto, CA 94301, December 1991.
- [15] Kurt Gödel. On formally undecidable propositions of *Principia Mathematica* and related systems I. In Jean van Heijenoort, editor, *From Frege to Gödel: A Source Book in Mathematical Logic*, pages 596–616. Harvard University Press, Cambridge, Massachusetts, 1967.
- [16] Douglas E. Harms. *The Influence of Software Reuse on Programming Language Design*. PhD thesis, Dept. of Computer and Information Science, The Ohio State University, Columbus, OH, 1990.
- [17] Douglas E. Harms and Bruce W. Weide. Copying and swapping: Influences on the design of reusable software components. *IEEE Transactions on Software Engineering*, 17(5):424–435, May 1991.
- [18] C. A. R. Hoare. An axiomatic basis for computer programming. *Communications of the ACM*, 12(10):576–580, 583, October 1969.
- [19] C. A. R. Hoare. Proof of correctness of data representation. *Acta Informatica*, 1(4):271–281, 1972.
- [20] C. A. R. Hoare. Proof of correctness of data representation. In David Gries, editor, *Programming Methodology: A Collection of Articles by Members of IFIP WG2.3*, chapter 19, pages 269–281. Springer-Verlag, 1978.
- [21] C. A. R. Hoare and P. E. Lauer. Consistent and complementary formal theories of the semantics of programming languages. *Acta Informatica*, 3:135–154, 1974.

- [22] Joseph E. Hollingsworth. *Software Component Design-for-Reuse: A Language-Independent Discipline Applied to Ada*. PhD thesis, Dept. of Computer and Information Science, The Ohio State University, Columbus, OH, 1992.
- [23] James Cornelius King. *A Program Verifier*. PhD thesis, Department of Computer Science, Carnegie-Mellon University, 1969.
- [24] Jürgen Koenemann and Scott P. Robertson. Expert problem solving strategies for program comprehension. In *Human Factors in Computing Systems: CHI-91: Reaching Through Technology: Proceedings of the Annual Conference on Human Factors in Computing Systems*, pages 125–130. Association for Computing Machinery, Addison-Wesley, May 1991.
- [25] Joan Krone. *The Role of Verification in Software Reusability*. PhD thesis, The Ohio State University, 1988.
- [26] Imre Lakatos. *Proofs and Refutations: The Logic of Mathematical Discovery*. Cambridge University Press, 1976. Edited by John Worrall and Elie Zahar.
- [27] Nancy G. Leveson and Clark S. Turner. An investigation of the Therac-25 accidents. *Computer*, 26(7):18–41, July 1993. IEEE Computer Society.
- [28] David C. Littman, Jeannine Pinto, Stanley Letovsky, and Elliot Soloway. Mental models and software maintenance. In Elliot Soloway and Sitharama Iyengar, editors, *Empirical Studies of Programmers: First Workshop*, Human/Computer Interaction, pages 80–98, Norwood, NJ, June 1986. Ablex Publishing.
- [29] Donald MacKenzie. The automation of proof: A historical and sociological exploration. *IEEE Annals of the History of Computing*, 17(3):7–29, Fall 1995.
- [30] Z. Manna. The correctness of programs. *Journal of Computer and Systems Sciences*, 3(2):119–127, 1969.
- [31] Z. Manna. Termination of programs represented as interpreted graphs. In *Joint Computer Conference*, volume 36, pages 83–89. American Federation of Information Processing Societies, Spring 1970.
- [32] Zohar Manna and Richard Waldinger. *The Logical Basis for Computer Programming*, volume 1, Deductive Reasoning. Addison-Wesley, Reading, Mass., 1985.

- [33] William Douglas Maurer. The modification index method of generating verification conditions. In *Proceedings of the 15th Annual ACM Southeast Regional Conference*, pages 426–440. Association for Computing Machinery, April 1977.
- [34] John McCarthy. Towards a mathematical science of computation. In C. M. Popplewell, editor, *Information Processing 1962: Proc. IFIP Congress 62*, pages 21–28, Amsterdam, 1963. North Holland.
- [35] R. Milne and C. Strachey. *A Theory of Programming Language Semantics*. Chapman & Hall Ltd., 1976.
- [36] Carroll Morgan. The specification statement. *ACM Transactions on Programming Languages and Systems*, 10(3):403–419, July 1988.
- [37] P. Naur. Proof of algorithms by general snapshots. *BIT*, 6:310–316, 1966.
- [38] Jainendra K. Navlakha. A syntax directed program verification system. Technical Report ESCI-77-3, Computer Engineering and Science Dept., Case Institute of Technology, Case Western Reserve University, Cleveland, Ohio 44106, February 1978.
- [39] Frank G. Pagan. *Formal Specification of Programming Languages: A Panoramic Primer*. Prentice-Hall Software Series. Prentice-Hall, 1981.
- [40] Roger S. Pressman. *Software Engineering: A Practitioner's Approach*. McGraw-Hill, third edition, 1992.
- [41] John M. Rushby and Friedrich von Henke. Formal verification of algorithms for critical systems. *IEEE Transactions on Software Engineering*, 19(1):13–23, January 1993.
- [42] John G. Sanderson. *A Relational Theory of Computing*, volume 82 of *Lecture Notes in Computer Science*. Springer-Verlag, New York, NY, 1980.
- [43] D. Scott and C. Strachey. Toward a mathematical semantics for computer languages. *Computers and Automation*, 1972.
- [44] Murali Sitaraman and Bruce W. Weide. Special feature: Component-based software using RESOLVE. *Software Engineering Notes*, 19(4):21–67, October 1994.
- [45] Muralidharan Sitaraman. *Mechanisms and Methods for Performance Tuning of Reusable Software Components*. PhD thesis, Dept. of Computer and Information Science, The Ohio State University, Columbus, OH, 1990.

- [46] R. L. Sites. *Proving That Computer Programs Terminate Cleanly*. PhD thesis, Department of Computer Science, Stanford University, May 1974.
- [47] Bruce W. Weide. Course notes for CIS 680: Software reuse through data structures and algorithms. Department of Computer and Information Science, The Ohio State University, January 1989.
- [48] Bruce W. Weide. Personal communication, August 1995.
- [49] Bruce W. Weide, William F. Ogden, and Stuart H. Zweben. Reusable software components. In Marshall C. Yovits, editor, *Advances in Computers*, volume 33, pages 1–65. Academic Press, 1991.