

- „A C Coding Standard,“ (1991): *Unix Review* 9, č. 9 (září): 42–43.
- Tarek K. Abdel-Hamid (1989): „*The Dynamics of Software Project Staffing: A System Dynamics Based Simulation Approach*,“ *IEEE Transactions on Software Engineering* SE-15, č. 2 (únor): 109–19.
- Abran, Alain, a kolektiv (2001): *Swebok: Guide to the Software Engineering Body of Knowledge: Trial Version 100-May 2001*. Los Alamitos, CA: IEEE Computer Society Press.
- Abrash, Michael (1992): „*Flooring It: The Optimization Challenge*,“ *PC Techniques* 2, č. 6 (únor/březen): 82–88.
- Ackerman, A. Frank, Lynne S. Buchwald a Frank H. Lewski (1989): „*Software Inspections: An Effective Verification Process*,“ *IEEE Software*, květen/červen 1989, 31–36.
- Adams, James L (2001): *Conceptual Blockbusting: A Guide to Better Ideas*, 4. vydání, Cambridge, MA: Perseus Publishing.
- Aho, Alfred V., Brian W. Kernighan a Peter J. Weinberg (1977): *The AWK Programming Language*. Reading, MA: Addison-Wesley.
- Aho, Alfred V., John E. Hopcroft a Jeffrey D. Ullman (1983): *Data Structures and Algorithms*. Reading, MA: Addison-Wesley.
- Albrecht, Allan J (1979): „*Measuring Application Development Productivity*,“ *Proceedings of the Joint SHARE/GUIDE/IBM Application Development Symposium*, říjen 1979: 83–92.
- Ambler, Scott (2003): *Agile Database Techniques*. New York, NY: John Wiley & Sons.
- Anand, N (1988): „*Clarify Function!*,“ *ACM Sigplan Notices* 23, č. 6 (červen): 69–79.
- Aristotle. *The Ethics of Aristotle: The Nicomachean Ethics*. Přeložil J.A.K. Thomson. Revize Hugh Tredennick. Vydáno v Hamondsworth, Middlesex, England: Penguin, 1976.
- Armenise, Pasquale (1989): „*A Structured Approach to Program Optimization*,“ *IEEE Transactions on Software Engineering* SE-15, č. 2 (únor): 101–8.
- Arnold, Ken, James Gosling a David Holmes (2000): *The Java Programming Language*, 3. vydání, Boston, MA: Addison-Wesley.
- Arthur, Lowell J (1988): *Software Evolution: The Software Maintenance Challenge*. New York, NY: John Wiley & Sons.
- Augustine, N. R (1979): „*Augustine's Laws and Major System Development Programs*,“ *Defense Systems Management Review*: 50–76.
- Babich, W (1986): *Software Configuration Management*. Reading, MA: Addison-Wesley.
- Bachman, Charles W (1973): „*The Programmer as Navigator*,“ Turing Award Lecture. *Communications of the ACM* 16, č. 11 (listopad): 653.
- Baecker, Ronald M. a Aaron Marcus (1990): *Human Factors and Typography for More Readable Programs*. Reading, MA: Addison-Wesley.
- Bairdain, E. F (1964): „*Research Studies of Programmers and Programming*,“ Unpublished studies reported in Boehm 1981.
- Baker, F. Terry a Harlan D. Mills (1973): „*Chief Programmer Teams*,“ *Datamation* 19, č. 12 (prosinec): 58–61.
- Barbour, Ian G (1966): *Issues in Science and Religion*. New York, NY: Harper & Row.
- Barbour, Ian G (1974): *Myths, Models a Paradigms: A Comparative Study in Science and Religion*. New York, NY: Harper & Row.
- Barwell, Fred a kolektiv (2002): *Professional VB.NET*, 2. vydání, Birmingham, UK: Wrox.
- Basili, V. R. a B. T. Perricone (1984): „*Software Errors and Complexity: An Empirical Investigation*,“ *Communications of the ACM* 27, č. 1 (leden): 42–52.
- Basili, Victor R. a Albert J. Turner (1975): „*Iterative Enhancement: A Practical Technique for Software Development*,“ *IEEE Transactions on Software Engineering* SE-1, č. 4 (prosinec): 390–96.
- Basili, Victor R. a David M. Weiss (1984): „*A Methodology for Collecting Valid Software Engineering Data*,“ *IEEE Transactions on Software Engineering* SE-10, č. 6 (listopad): 728–38.
- Basili, Victor R. a Richard W. Selby (1987): „*Comparing the Effectiveness of Software Testing Strategies*,“ *IEEE Transactions on Software Engineering* SE-13, č. 12 (prosinec): 1278–96.
- Basili, Victor R. a kolektiv (2002): „*Lessons learned from 25 years of process improvement: The Rise and Fall of the NASA Software Engineering Laboratory*,“ *Proceedings of the 24th International Conference on Software Engineering*, Orlando, FL.
- Basili, Victor R., Richard W. Selby a David H. Hutchens (1986): „*Experimentation in Software Engineering*,“ *IEEE Transactions on Software Engineering* SE-12, č. 7 (July): 733–43.

- Basili, Victor, L. Briand a W.L. Melo (1996): „*A Validation of Object-Oriented Design Metrics as Quality Indicators*,“ IEEE Transactions on Software Engineering, říjen 1996, 751–761.
- Bass, Len, Paul Clements a Rick Kazman (2003): *Software Architecture in Practice*, 2. vydání Boston, MA: Addison-Wesley.
- Bastani, Farokh a Sitharama Iyengar (1987): „*The Effect of Data Structures on the Logical Complexity of Programs*,“ Communications of the ACM 30, č. 3 (březen): 250–59.
- Bays, Michael (1999): *Software Release Methodology*. Englewood Cliffs, NJ: Prentice Hall.
- Beck, Kent (2000): *Extreme Programming Explained: Embrace Change*. Reading, MA: Addison-Wesley.
- Beck, Kent (2003): *Test-Driven Development: By Example*. Boston, MA: Addison-Wesley.
- Beck, Kent (1991): „*Think Like An Object*,“ Unix Review 9, č. 10 (říjen): 39–43.
- Beck, Leland L. a Thomas E. Perkins (1983): „*A Survey of Software Engineering Practice: Tools, Methods a Results*,“ IEEE Transactions on Software Engineering SE-9, č. 5 (září): 541–61.
- Beizer, Boris (1990): *Software Testing Techniques*, 2. vydání, New York, NY: Van Nostrand Reinhold.
- Bentley, Jon a Donald Knuth (1986): „*Literate Programming*,“ Communications of the ACM 29, č. 5 (květen): 364–69.
- Bentley, Jon, Donald Knuth a Doug McIlroy (1986): „*A Literate Program*,“ Communications of the ACM 29, č. 5 (květen): 471–83.
- Bentley, Jon (1982): *Writing Efficient Programs*. Englewood Cliffs, NJ: Prentice Hall.
- Bentley, Jon (1988): *More Programming Pearls: Confessions of a Coder*. Reading, MA: Addison-Wesley.
- Bentley, Jon (1991): „*Software Exploratorium: Writing Efficient C Programs*,“ Unix Review 9, č. 8 (srpen): 62–73.
- Bentley, Jon (2000): *Programming Pearls*, 2. vydání, Reading, MA: Addison-Wesley.
- Berczuk, Stephen P. a Brad Appleton (2003): *Software Configuration Management Patterns: Effective Teamwork, Practical Integration*. Boston, MA: Addison-Wesley.
- Berry, R. E. a B. A. E. Meekings (1985): „*A Style Analysis of C Programs*,“ Communications of the ACM 28, č. 1 (leden): 80–88.
- Bersoff, Edward H (1984): „*Elements of Software Configuration Management*,“ IEEE Transactions on Software Engineering SE-10, č. 1 (leden): 79–87.
- Bersoff, Edward H. a Alan M. Davis (1991): „*Impacts of Life Cycle Models on Software Configuration Management*,“ Communications of the ACM 34, č. 8 (srpen): 104–18.
- Bersoff, Edward H. a kolektiv (1980): *Software Configuration Management*. Englewood Cliffs, NJ: Prentice Hall.
- Birrell, N. D. a M. A. Ould (1985): *A Practical Handbook for Software Development*. Cambridge, England: Cambridge University Press.
- Bloch, Joshua (2001): *Effective Java Programming Language Guide*. Boston, MA: Addison-Wesley.
- BLS (2002): *Occupational Outlook Handbook*, vydání, 2002–03, Bureau of Labor Statistics.
- BLS (2004): *Occupational Outlook Handbook*, vydání, 2004–05, Bureau of Labor Statistics.
- Blum, Bruce I (1989): „*A Software Environment: Some Surprising Empirical Results*,“ Zápis ze 14. výročního semináře o softwarovém inženýrství, listopad 29, 1989. Greenbelt, MD: Goddardovo centrum kosmických letů v NASA. Dokument SEL-89-007.
- Boddie, John (1987): *Crunch Mode*. New York, NY: Yourdon Press.
- Boehm, Barry a Richard Turner (2004): *Balancing Agility and Discipline: A Guide for the Perplexed*. Boston, MA: Addison-Wesley.
- Boehm, Barry W (1981): *Software Engineering Economics*. Englewood Cliffs, NJ: Prentice Hall.
- Boehm, Barry W (1984): „*Software Engineering Economics*,“ IEEE Transactions on Software Engineering SE-10, č. 1 (leden): 4–21.
- Boehm, Barry W. (1987a): „*Improving Software Productivity*,“ IEEE Computer, září, 43–57.
- Boehm, Barry W. (1987b): „*Industrial Software Metrics Top 10 List*,“ IEEE Software 4, č. 9 (září): 84–85.
- Boehm, Barry W (1988): „*A Spiral Model of Software Development and Enhancement*,“ Computer, květen, 61–72.
- Boehm, Barry W. a Philip N. Papaccio (1988): „*Understanding and Controlling Software Costs*,“ IEEE Transactions on Software Engineering SE-14, č. 10 (říjen): 1462–77.
- Boehm, Barry W., ed (1989): *Tutorial: Software Risk Management*. Washington, DC: IEEE Computer Society Press.

- Boehm, Barry W. a kolektiv (1978): *Characteristics of Software Quality*. New York, NY: North-Holland.
- Boehm, Barry W. a kolektiv (1984): „A Software Development Environment for Improving Productivity,” *Computer*, červen, 30–44.
- Boehm, Barry W., T. E. Gray a T. Seewaldt (1984): „Prototyping Versus Specifying: A Multiproject Experiment,” *IEEE Transactions on Software Engineering* SE-10, č. 3 (květen): 290–303. Also in Jones 1986b.
- Boehm, Barry a kolektiv (2000a): *Software Cost Estimation with Cocomo II*. Boston, MA: Addison-Wesley.
- Boehm, Barry (2000b): „Unifying Software Engineering and Systems Engineering,” *IEEE Computer*, březen 2000, 114–116.
- Boehm-Davis, Deborah, Sylvia Sheppard a John Bailey (1987): „Program Design Languages: How Much Detail Should They Include?,” *International Journal of Man-Machine Studies* 27, č. 4: 337–47.
- Böhm, C. a G. Jacopini (1966): „Flow Diagrams, Turing Machines and Languages with Only Two Formation Rules,” *Communications of the ACM* 9, č. 5 (květen): 366–71.
- Booch, Grady (1987): *Software Engineering with Ada*, 2. vydání, Menlo Park, CA: Benjamin/Cummings.
- Booch, Grady (1994): *Object Oriented Analysis and Design with Applications*, 2. vydání, Boston, MA: Addison-Wesley.
- Booth, Rick (1997): *Inner Loops: A Sourcebook for Fast 32-bit Software Development*. Boston, MA: Addison-Wesley.
- Boundy, David (1991): „A Taxonomy of Programmers,” *ACM SIGSOFT Software Engineering Notes* 16, č. 4 (říjen): 23–30.
- Brand, Stewart (1995): *How Buildings Learn: What Happens After They're Built*. Penguin USA.
- Branstad, Martha A., John C. Cherniavsky a W. Richards Adrion (1980): „Validation, Verification a Testing for the Individual Programmer,” *Computer*, prosinec, 24–30.
- Brockmann, R. John (1990): *Writing Better Computer User Documentation: From Paper to Hypertext: Version 2.0*. New York, NY: John Wiley & Sons.
- Brooks, Frederick P., Jr (1987): „No Silver Bullets – Essence and Accidents of Software Engineering,” *Computer*, duben, 10–19.
- Brooks, Frederick P., Jr (1995): *The Mythical Man-Month: Essays on Software Engineering, Anniversary Edition* (2. vydání). Reading, MA: Addison-Wesley.
- Brooks, Ruven (1977): „Towards a Theory of the Cognitive Processes in Computer Programming,” *International Journal of Man-Machine Studies* 9:737–51.
- Brooks, W. Douglas (1981): „Software Technology Payoff – Some Statistical Evidence,” *The Journal of Systems and Software* 2:3–9.
- Brown, A. R. a W. A. Sampson (1973): *Program Debugging*. New York, NY: American Elsevier.
- Buschman, Frank a kolektiv (1996): *Pattern-Oriented Software Architecture, Volume 1: A System of Patterns*. New York, NY: John Wiley & Sons.
- Bush, Marilyn a John Kelly (1989): „The Jet Propulsion Laboratory's Experience with Formal Inspections,” zápis ze 14. výročního semináře o softwarovém inženýrství, listopad 29, 1989. Greenbelt, MD: Goddardovo centrum kosmických letů v NASA. Dokument SEL-89-007.
- Caine, S. H. a E. K. Gordon (1975): „PDL – A Tool for Software Design,” zápis z národní počítačové konference z roku 1975 v Montvale, NJ: AFIPS Press, 271–76.
- Card, David N (1987): „A Software Technology Evaluation Program,” *Information and Software Technology* 29, č. 6 (July/srpen): 291–300.
- Card, David N., Frank E. McGarry a Gerald T. Page (1987): „Evaluating Software Engineering Technologies,” *IEEE Transactions on Software Engineering* SE-13, č. 7 (July): 845–51.
- Card, David N., Victor E. Church a William W. Agresti (1986): „An Empirical Study of Software Design Practices,” *IEEE Transactions on Software Engineering* SE-12, č. 2 (únor): 264–71.
- Card, David N., with Robert L. Glass (1990): *Measuring Software Design Quality*. Englewood Cliffs, NJ: Prentice Hall.
- Card, David, Gerald Page a Frank Mc-Garry (1985): „Criteria for Software Modularization,” zápis z 8. mezinárodní konference o softwarovém inženýrství. Washington, DC: IEEE Computer Society Press, 372–77.
- Carnegie, Dale (1981): *How to Win Friends and Influence People, Revised Edition*. New York, NY: Pocket Books.
- Chase, William G. a Herbert A. Simon (1973): „Perception in Chess,” *Cognitive Psychology* 4:55–81.
- Clark, R. Lawrence (1973): „A Linguistic Contribution of GOTO-less Programming,” *Datamation*, prosinec 1973.

- Clements, Paul, ed (2003): *Documenting Software Architectures: Views and Beyond*. Boston, MA: Addison-Wesley.
- Clements, Paul, Rick Kazman a Mark Klein (2002): *Evaluating Software Architectures: Methods and Case Studies*. Boston, MA: Addison-Wesley.
- Coad, Peter a Edward Yourdon (1991): *Object-Oriented Design*. Englewood Cliffs, NJ: Yourdon Press.
- Cobb, Richard H. a Harlan D. Mills (1990): „Engineering Software Under Statistical Quality Control,“ *IEEE Software* 7, č. 6 (listopad): 45–54.
- Cockburn, Alistair (2000): *Writing Effective Use Cases*. Boston, MA: Addison-Wesley.
- Cockburn, Alistair (2002): *Agile Software Development*. Boston, MA: Addison-Wesley.
- Collofello, Jim a Scott Woodfield (1989): „Evaluating the Effectiveness of Reliability Assurance Techniques,“ *Journal of Systems and Software* 9, č. 3 (březen).
- Comer, Douglas (1981): „Principles of Program Design Induced from Experience with Small Public Programs,“ *IEEE Transactions on Software Engineering* SE-7, č. 2 (březen): 169–74.
- Constantine, Larry L. (1990a): „Comments on ‘On Criteria for Module Interfaces’,“ *IEEE Transactions on Software Engineering* SE-16, č. 12 (prosinec): 1440.
- Constantine, Larry L. (1990b): „Objects, Functions a Program Extensibility,“ *Computer Language*, leden, 34–56.
- Conte, S. D., H. E. Dunsmore a V. Y. Shen (1986): *Software Engineering Metrics and Models*. Menlo Park, CA: Benjamin/Cummings.
- Cooper, Doug a Michael Clancy (1982): *Oh! Pascal!* 2. vydání, New York, NY: Norton.
- Cooper, Kenneth a Thomas Mullen (1993): „Swords and Plowshares: The Rework Cycles of Defense and Commercial Software Development Projects,“ *American Programmer*, květen 1993, 41–51.
- Corbató, Fernando J. (1991): „On Building Systems That Will Fail,“ 1991 Turing Award Lecture. *Communications of the ACM* 34, č. 9 (září): 72–81.
- Cornell, Gary a Jonathan Morrison (2002): *Programming VB .NET: A Guide for Experienced Programmers*, Berkeley, CA: Apress.
- Corwin, Al (1991): Private communication. CSTB (1990): „Scaling Up: A Research Agenda for Software Engineering,“ Excerpts from a report by the Computer Science and Technology Board. *Communications of the ACM* 33, č. 3 (březen): 281–93.
- Curtis, Bill, ed (1985): *Tutorial: Human Factors in Software Development*. Los Angeles, CA: IEEE Computer Society Press.
- Curtis, Bill a kolektiv (1986): „Software Psychology: The Need for an Interdisciplinary Program,“ , zápis z IEEE 74, č. 8: 1092–1106.
- Curtis, Bill a kolektiv (1989): „Experimentation of Software Documentation Formats,“ *Journal of Systems and Software* 9, č. 2 (únor): 167–207.
- Curtis, Bill, H. Krasner a N. Iscoe (1988): „A Field Study of the Software Design Process for Large Systems,“ *Communications of the ACM* 31, č. 11 (listopad): 1268–87.
- Curtis, Bill (1981): „Substantiating Programmer Variability,“ zápis z IEEE 69, č. 7: 846.
- Cusumano, Michael a Richard W. Selby (1995): *Microsoft Secrets*. New York, NY: The Free Press.
- Cusumano, Michael a kolektiv (2003): „Software Development Worldwide: The State of the Practice,“ *IEEE Software*, listopad/prosinec 2003, 28–34.
- Dahl, O. J., E. W. Dijkstra a C. A. R. Hoare (1972): *Structured Programming*. New York, NY: Academic Press.
- Date, Chris (1977): *An Introduction to Database Systems*. Reading, MA: Addison-Wesley.
- Davidson, Jack W. a Anne M. Holler (1992): „Subprogram Inlining: A Study of Its Effects on Program Execution Time,“ *IEEE Transactions on Software Engineering* SE-18, č. 2 (únor): 89–102.
- Davis, P. J. (1972): „Fidelity in Mathematical Discourse: Is One and One Really Two?,“ *American Mathematical Monthly*, březen, 252–63.
- DeGrace, Peter a Leslie Stahl (1990): *Wicked Problems, Righteous Solutions: A Catalogue of Modern Software Engineering Paradigms*. Englewood Cliffs, NJ: Yourdon Press.
- DeMarco, Tom a Timothy Lister (1999): *Peopleware: Productive Projects and Teams*, 2. vydání, New York, NY: Dorset House.
- DeMarco, Tom a Timothy Lister (1985): „Programmer Performance and the Effects of the Workplace,“ zápis z 8. mezinárodní konference o softwarovém inženýrství. Washington, DC: IEEE Computer Society Press, 268–72.
- DeMarco, Tom (1979): *Structured Analysis and Systems Specification: Tools and Techniques*. Englewood Cliffs, NJ: Prentice Hall.
- DeMarco, Tom (1982): *Controlling Software Projects*. New York, NY: Yourdon Press.

- DeMillo, Richard A., Richard J. Lipton a Alan J. Perlis (1979): „*Social Processes and Proofs of Theorems and Programs*,“ Communications of the ACM 22, č. 5 (květen): 271–80.
- Dijkstra, Edsger (1965): „*Programming Considered as a Human Activity*,“ zápis z kongresu IFIP v roce 1965. Amsterdam: North-Holland, 213–17. Reprinted in Yourdon 1982.
- Dijkstra, Edsger (1968): „*Go To Statement Considered Harmful*,“ Communications of the ACM 11, č. 3 (březen): 147–48.
- Dijkstra, Edsger (1969): „*Structured Programming*,“ Reprinted in Yourdon 1979.
- Dijkstra, Edsger (1972): „*The Humble Programmer*,“ Communications of the ACM 15, č. 10 (říjen): 859–66.
- Dijkstra, Edsger (1985): „*Fruits of Misunderstanding*,“ Datamation, únor 15, 86–87.
- Dijkstra, Edsger (1989): „*On the Cruelty of Really Teaching Computer Science*,“ Communications of the ACM 32, č. 12 (prosinec): 1397–1414.
- Dunn, Robert H. (1984): *Software Defect Removal*. New York, NY: McGraw-Hill.
- Ellis, Margaret A. a Bjarne Stroustrup (1990): *The Annotated C++ Reference Manual*. Boston, MA: Addison-Wesley.
- Elmasri, Ramez a Shamkant B. Navathe (1989): *Fundamentals of Database Systems*. Redwood City, CA: Benjamin/Cummings.
- Elshoff, James L. (1976): „*An Analysis of Some Commercial PL/I Programs*,“ IEEE Transactions on Software Engineering SE-2, č. 2 (červen): 113–20.
- Elshoff, James L. (1977): „*The Influence of Structured Programming on PL/I Program Profiles*,“ IEEE Transactions on Software Engineering SE-3, č. 5 (září): 364–68.
- Elshoff, James L. a Michael Marcotty (1982): „*Improving Computer Program Readability to Aid Modification*,“ Communications of the ACM 25, č. 8 (srpen): 512–21.
- Endres, Albert (1975): „*An Analysis of Errors and Their Causes in System Programs*,“ IEEE Transactions on Software Engineering SE-1, č. 2 (červen): 140–49.
- Evangelist, Michael (1984): „*Program Complexity and Programming Style*,“ zápis z první mezinárodní konference o datovém inženýrství. New York, NY: IEEE Computer Society Press, 534–41.
- Fagan, Michael E. (1976): „*Design and Code Inspections to Reduce Errors in Program Development*,“ IBM Systems Journal 15, č. 3: 182–211.
- Fagan, Michael E. (1986): „*Advances in Software Inspections*,“ IEEE Transactions on Software Engineering SE-12, č. 7 (July): 744–51.
- Federal Software Management Support Center (1986): *Programmers Work-bench Handbook*. Falls Church, VA: Office of Software Development and Information Technology.
- Feiman, J. a M. Driver (2002): „*Leading Programming Languages for IT Portfolio Planning*,“ Gartner Research report SPA-17-6636, září 27, 2002.
- Fetzer, James H. (1988): „*Program Verification: The Very Idea*,“ Communications of the ACM 31, č. 9 (září): 1048–63.
- FIPS PUB 38, *Guidelines for Documentation of Computer Programs and Automated Data Systems* (1976), ministerstvo obchodu Spojených států, národní úřad pro standardy. Washington, DC: U.S. Government Printing Office, Feb. 15.
- Fishman, Charles (1996): „*They Write the Right Stuff*,“ Fast Company, prosinec 1996.
- Fjelstad, R. K. a W. T. Hamlen (1979): „*Applications Program Maintenance Study: Report to our Respondents*,“ průvodce jednáním 48, Philadelphia. Reprinted in Tutorial on Software Maintenance, G. Parikh and N. Zvegintzov, eds. Los Alamitos, CA: CS Press, 1983: 13–27.
- Floyd, Robert (1979): „*The Paradigms of Programming*,“ Communications of the ACM 22, č. 8 (srpen): 455–60.
- Fowler, Martin (1999): *Refactoring: Improving the Design of Existing Code*. Reading, MA: Addison-Wesley.
- Fowler, Martin (2002): *Patterns of Enterprise Application Architecture*. Boston, MA: Addison-Wesley.
- Fowler, Martin (2003): *UML Distilled: A Brief Guide to the Standard Object Modeling Language*, 3. vydání, Boston, MA: Addison-Wesley.
- Fowler, Martin (2004): *UML Distilled*, 3. vydání Boston, MA: Addison-Wesley.
- Fowler, Priscilla J. (1986): „*In-Process Inspections of Work Products at AT&T*,“ AT&T Technical Journal, březen/duben, 102–12.
- Foxall, James (2003): *Practical Standards for Microsoft Visual Basic .NET*. Redmond, WA: Microsoft Press.
- Freedman, Daniel P. a Gerald M. Weinberg (1990): *Handbook of Walkthroughs, Inspections and*

- Technical Reviews*, 3. vydání, New York, NY: Dorset House.
- Freeman, Peter a Anthony I. Wasserman, eds. (1983): *Tutorial on Software Design Techniques*, 4. vydání, Silver Spring, MD: IEEE Computer Society Press.
- Gamma, Erich a kolektiv (1995): *Design Patterns*. Reading, MA: Addison-Wesley.
- Gerber, Richard (2002): *Software Optimization Cookbook: High-Performance Recipes for the Intel Architecture*. Intel Press.
- Gibson, Elizabeth (1990): „Objects – Born and Bred,“ *BYTE*, říjen, 245–54.
- Gilb, Tom a Dorothy Graham (1993): *Software Inspection*. Wokingham, England: Addison-Wesley.
- Gilb, Tom (1977): *Software Metrics*. Cambridge, MA: Winthrop.
- Gilb, Tom (1988): *Principles of Software Engineering Management*. Wokingham, England: Addison-Wesley.
- Gilb, Tom (2004): *Competitive Engineering*. Boston, MA: Addison-Wesley. Ke stažení na adrese www.result-planning.com.
- Ginac, Frank P. (1998): *Customer Oriented Software Quality Assurance*. Englewood Cliffs, NJ: Prentice Hall.
- Glass, Robert L. (1982): *Modern Programming Practices: A Report from Industry*. Englewood Cliffs, NJ: Prentice Hall.
- Glass, Robert L. (1988): *Software Communication Skills*. Englewood Cliffs, NJ: Prentice Hall.
- Glass, Robert L. (1991): *Software Conflict: Essays on the Art and Science of Software Engineering*. Englewood Cliffs, NJ: Yourdon Press.
- Glass, Robert L. (1995): *Software Creativity*. Reading, MA: Addison-Wesley.
- Glass, Robert L. (1999): „Inspections – Some Surprising Findings,“ *Communications of the ACM*, duben 1999, 17–19.
- Glass, Robert L. (1999): „The realities of software technology payoffs,“ *Communications of the ACM*, únor 1999, 74–79.
- Glass, Robert L. (2003): *Facts and Fallacies of Software Engineering*. Boston, MA: Addison-Wesley.
- Glass, Robert L. a Ronald A. Noiseux (1981): *Software Maintenance Guidebook*. Englewood Cliffs, NJ: Prentice Hall.
- Gordon, Ronald D. (1979): „Measuring Improvements in Program Clarity,“ *IEEE Transactions on Software Engineering* SE-5, č. 2 (březen): 79–90.
- Gordon, Scott V. a James M. Bieman (1991): „Rapid Prototyping and Software Quality: Lessons from Industry,“ Ninth Annual Pacific Northwest Software Quality Conference, říjen 7–8. Oregon Convention Center, Portland, OR.
- Gorla, N., A. C. Benander a B. A. Benander (1990): „Debugging Effort Estimation Using Software Metrics,“ *IEEE Transactions on Software Engineering* SE-16, č. 2 (únor): 223–31.
- Gould, John D. (1975): „Some Psychological Evidence on How People Debug Computer Programs,“ *International Journal of Man-Machine Studies* 7:151–82.
- Grady, Robert B. (1987): „Measuring and Managing Software Maintenance,“ *IEEE Software* 4, č. 9 (zář): 34–45.
- Grady, Robert B. (1993): „Practical Rules of Thumb for Software Managers,“ *The Software Practitioner* 3, č. 1 (leden/únor): 4–6.
- Grady, Robert B. (1999): „An Economic Release Decision Model: Insights into Software Project Management,“ v zápisu z konference na téma uplatnění měření v softwaru, 227–239. Orange Park, FL: Software Quality Engineering.
- Grady, Robert B. a Tom Van Slack (1994): „Key Lessons in Achieving Widespread Inspection Use,“ *IEEE Software*, July 1994.
- Grady, Robert B. (1992): *Practical Software Metrics For Project Management And Process Improvement*. Englewood Cliffs, NJ: Prentice Hall.
- Grady, Robert B. a Deborah L. Caswell (1987): *Software Metrics: Establishing a Company-Wide Program*. Englewood Cliffs, NJ: Prentice Hall.
- Green, Paul (1987): „Human Factors in Computer Systems, Some Useful Readings,“ *Sigchi Bulletin* 19, č. 2: 15–20.
- Gremillion, Lee L. (1984): „Determinants of Program Repair Maintenance Requirements,“ *Communications of the ACM* 27, č. 8 (srpen): 826–32.
- Gries, David (1981): *The Science of Programming*. New York, NY: Springer-Verlag.
- Grove, Andrew S. (1983): *High Output Management*. New York, NY: Random House.
- Haley, Thomas J. (1996): „Software Process Improvement at Raytheon,“ *IEEE Software*, listopad 1996.
- Hansen, John C. a Roger Yim (1987): „Indentation Styles in C,“ *SIGSMALL/PC Notes* 13, č. 3 (srpen): 20–23.
- Hanson, Dines (1984): *Up and Running*. New York, NY: Yourdon Press.

- Harrison, Warren a Curtis Cook (1986): „Are Deeply Nested Conditionals Less Readable?“, *Journal of Systems and Software* 6, č. 4 (listopad): 335–42.
- Hasan, Jeffrey a Kenneth Tu (2003): *Performance Tuning and Optimizing ASP.NET Applications*. Apress.
- Hass, Anne Mette Jonassen (2003): *Configuration Management Principles and Practices*. Boston, MA: Addison-Wesley.
- Hatley, Derek J. a Imtiaz A. Pirbhai (1988): *Strategies for Real-Time System Specification*. New York, NY: Dorset House.
- Hecht, Alan (1990): „Cute Object-oriented Acronyms Considered FOolish“, *Software Engineering Notes*, leden, 48.
- Heckel, Paul (1994): *The Elements of Friendly Software Design*. Alameda, CA: Sybex.
- Hecker, Daniel E. (2001): „Occupational Employment Projections to 2010“, *Monthly Labor Review*, listopad 2001.
- Hecker, Daniel E. (2004): „Occupational Employment Projections to 2012“, *Monthly Labor Review*, únor 2004, Vol. 127, č. 2, str. 80–105.
- Henry, Sallie a Dennis Kafura (1984): „The Evaluation of Software Systems' Structure Using Quantitative Software Metrics“, *Software – Practice and Experience* 14, č. 6 (červen): 561–73.
- Hetzel, Bill (1988): *The Complete Guide to Software Testing*, 2. vydání, Wellesley, MA: QED Information Systems.
- Highsmith, James A., III. (2000): *Adaptive Software Development: A Collaborative Approach to Managing Complex Systems*. New York, NY: Dorset House.
- Highsmith, Jim (2002): *Agile Software Development Ecosystems*. Boston, MA: Addison-Wesley.
- Hildebrand, J. D. (1989): „An Engineer's Approach“, *Computer Language*, říjen, 5–7.
- Hoare, Charles Anthony Richard, 1981. „The Emperor's Old Clothes“, *Communications of the ACM*, únor 1981, 75–83.
- Hollocker, Charles P. (1990): *Software Reviews and Audits Handbook*. New York, NY: John Wiley & Sons.
- Houghton, Raymond C. (1990): „An Office Library for Software Engineering Professionals“, *Software Engineering: Tools, Techniques, Practice*, květen/červen, 35–38.
- Howard, Michael a David LeBlanc (2003): *Writing Secure Code*, 2. vydání Redmond, WA: Microsoft Press.
- Hughes, Charles E., Charles P. Pfleeger a Lawrence L. Rose (1978): *Advanced Programming Techniques: A Second Course in Programming Using Fortran*. New York, NY: John Wiley & Sons.
- Humphrey, Watts S. (1989): *Managing the Software Process*. Reading, MA: Addison-Wesley.
- Humphrey, Watts S. (1995): *A Discipline for Software Engineering*. Reading, MA: Addison-Wesley.
- Humphrey, Watts S., Terry R. Snyder a Ronald R. Willis (1991): „Software Process Improvement at Hughes Aircraft“, *IEEE Software* 8, č. 4 (July): 11–23.
- Humphrey, Watts (1997): *Introduction to the Personal Software Process*. Reading, MA: Addison-Wesley.
- Humphrey, Watts (2002): *Winning with Software: An Executive Strategy*. Boston, MA: Addison-Wesley.
- Hunt, Andrew a David Thomas (2000): *The Pragmatic Programmer*. Boston, MA: Addison-Wesley.
- Ichbiah, Jean D. a kolektiv (1986): *Rationale for Design of the Ada Programming Language*. Minneapolis, MN: Honeywell Systems and Research Center.
- IEEE Software 7, č. 3 (květen 1990).
- IEEE Std 1008-1987 (R1993), *Standard pro testování jednotek*
- IEEE Std 1016-1998, *Doporučený postup pro popis softwarového návrhu*
- IEEE Std 1028-1997, *Standard pro softwarové revize*
- IEEE Std 1045-1992, *Standard pro měření softwarové produktivity*
- IEEE Std 1058-1998, *Standard pro plány řízení softwarových projektů*
- IEEE Std 1061-1998, *Standard pro metodiku měření kvality softwaru*
- IEEE Std 1062-1998, *Doporučený postup pro pořizování softwaru*
- IEEE Std 1063-2001, *Standard pro uživatelskou softwarovou dokumentaci*
- IEEE Std 1074-1997, *Standard pro procesy životního cyklu vývoje softwaru*
- IEEE Std 1219-1998, *Standard pro údržbu softwaru*
- IEEE Std 1233-1998, *Guide pro vývoj specifikací systémových požadavků*
- IEEE Std 1233-1998. *Průvodce IEEE vývojem specifikací systémových požadavků*
- IEEE Std 1471-2000. *Doporučený postup pro architektonický popis softwarových systémů*

- IEEE Std 1490-1998, *Průvodce – Přijetí standardu PMI – Průvodce řízením projektu*
- IEEE Std 1540-2001, *Standard pro procesy životního cyklu softwaru – řízení rizik*
- IEEE Std 730-2002, *Standard pro plány zajištění kvality softwaru*
- IEEE Std 828-1998, *Standard pro plány řízení softwarové konfigurace*
- IEEE Std 829-1998, *Standard pro testovací dokumentaci softwaru*
- IEEE Std 830-1998, *Doporučený postup pro specifikace softwarových požadavků*
- IEEE Std 830-1998. *IEEE Recommended Practice for Software Requirements Specifications*. Los Alamitos, CA: IEEE Computer Society Press.
- IEEE, 1991. *IEEE Software Engineering Standards Collection, Spring 1991 Edition*. New York, NY: Institute of Electrical and Electronics Engineers.
- IEEE, 1992. „Rear Adm. Grace Hopper dies at 85,“ *IEEE Computer*, únor, 84.
- Ingrassia, Frank S. (1976): „*The Unit Development Folder (UDF): An Effective Management Tool for Software Development*,“ TRW Technical Report TRW-SS-76-11. Also reprinted in Reifer 1986, 366–79.
- Ingrassia, Frank S. (1987): „*The Unit Development Folder (UDF): A Ten-Year Perspective*,“ Tutorial: Software Engineering Project Management, ed. Richard H. Thayer. Los Alamitos, CA: IEEE Computer Society Press, 405–15.
- Jackson, Michael A. (1975): *Principles of Program Design*. New York, NY: Academic Press.
- Jacobson, Ivar, Grady Booch a James Rumbaugh (1999): *The Unified Software Development Process*. Reading, MA: Addison-Wesley.
- Johnson, Jim (1999): „*Turning Chaos into Success*,“ *Software Magazine*, prosinec 1999, 30–39.
- Johnson, Mark (1994a): „*Dr. Boris Beizer on Software Testing: An Interview Part 1*,“ *The Software QA Quarterly*, Spring 1994, 7–13.
- Johnson, Mark (1994b): „*Dr. Boris Beizer on Software Testing: An Interview Part 2*,“ *The Software QA Quarterly*, Summer 1994, 41–45.
- Johnson, Walter L. (1987): „*Some Comments on Coding Practice*,“ *ACM SIGSOFT Software Engineering Notes* 12, č. 2 (duben): 32–35.
- Jones, T. Capers (1977): „*Program Quality and Programmer Productivity*,“ *IBM Technical Report TR 02.764*, leden, 42–78. Also in Jones 1986b.
- Jones, Capers (1986a): *Programming Productivity*. New York, NY: McGraw-Hill.
- Jones, T. Capers, ed (1986b): *Tutorial: Programming Productivity: Issues for the Eighties*, 2. vydání, Los Angeles, CA: IEEE Computer Society Press.
- Jones, Capers (1996): „*Software Defect-Removal Efficiency*,“ *IEEE Computer*, duben 1996.
- Jones, Capers (1997): *Applied Software Measurement: Assuring Productivity and Quality*, 2. vydání, New York, NY: McGraw-Hill.
- Jones, Capers (1998): *Estimating Software Costs*. New York, NY: McGraw-Hill.
- Jones, Capers (2000): *Software Assessments, Benchmarks a Best Practices*. Reading, MA: Addison-Wesley.
- Jones, Capers (2003): „*Variations in Software Development Practices*,“ *IEEE Software*, listopad/prosinec 2003, 22–27.
- Jonsson, Dan (1989): „*Next: The Elimination of GoTo-Patches?*,“ *ACM Sigplan Notices* 24, č. 3 (březen): 85–92.
- Kaelbling, Michael (1988): „*Programming Languages Should NOT Have Comment Statements*,“ *ACM Sigplan Notices* 23, č. 10 (říjen): 59–60.
- Kaner, Cem, Jack Falk a Hung Q. Nguyen (1999): *Testing Computer Software*, 2. vydání New York, NY: John Wiley & Sons.
- Kaner, Cem, James Bach a Bret Pettichord (2002): *Lessons Learned in Software Testing*. New York, NY: John Wiley & Sons.
- Keller, Daniel (1990): „*A Guide to Natural Naming*,“ *ACM Sigplan Notices* 25, č. 5 (květen): 95–102.
- Kelly, John C. (1987): „*A Comparison of Four Design Methods for Real-Time Systems*,“ zápis z 9. mezinárodní konference o softwarovém inženýrství. 238–52.
- Kelly-Bootle, Stan (1981): *The Devil's DP Dictionary*. New York, NY: McGraw-Hill.
- Kernighan, Brian W. a Rob Pike (1999): *The Practice of Programming*. Reading, MA: Addison-Wesley.
- Kernighan, Brian W. a P. J. Plauger (1976): *Software Tools*. Reading, MA: Addison-Wesley.
- Kernighan, Brian W. a P. J. Plauger (1978): *The Elements of Programming Style*. 2. vydání New York, NY: McGraw-Hill.
- Kernighan, Brian W. a P. J. Plauger (1981): *Software Tools in Pascal*. Reading, MA: Addison-Wesley.
- Kernighan, Brian W. a Dennis M. Ritchie (1988): *The C Programming Language*, 2. vydání, Englewood Cliffs, NJ: Prentice Hall.

- Killelea, Patrick (2002): *Web Performance Tuning*, 2. vydání Sebastopol, CA: O'Reilly & Associates.
- King, David (1988): *Creating Effective Software: Computer Program Design Using the Jackson Methodology*. New York, NY: Yourdon Press.
- Knuth, Donald (1971): „An Empirical Study of FORTRAN programs,“ *Software – Practice and Experience* 1:105–33.
- Knuth, Donald (1974): „Structured Programming with go to Statements,“ v *Classics in Software Engineering*, s redakcí Edwarda Yourdona. Englewood Cliffs, NJ: Yourdon Press, 1979.
- Knuth, Donald (1986): *Computers and Typesetting, Volume B, TEX: The Program*. Reading, MA: Addison-Wesley.
- Knuth, Donald (1997a): *The Art of Computer Programming, vol 1, Fundamental Algorithms*, 3. vydání, Reading, MA: Addison-Wesley.
- Knuth, Donald (1997b): *The Art of Computer Programming, vol. 2, Seminumerical Algorithms*, 3. vydání, Reading, MA: Addison-Wesley.
- Knuth, Donald (1998): *The Art of Computer Programming, vol. 3, Sorting and Searching*, 2. vydání, Reading, MA: Addison-Wesley.
- Knuth, Donald (2001): *Literate Programming*. Cambridge University Press.
- Korson, Timothy D. a Vijay K. Vaishnavi (1986): „An Empirical Study of Modularity on Program Modifiability,“ In Soloway and Iyengar 1986: 168–86.
- Kouchakdjian, Ara, Scott Green a Victor Basili (1989): „Evaluation of the Cleanroom Methodology in the Software Engineering Laboratory,“ zápis z 14. výročního semináře na téma softwarového inženýrství, listopad 29, 1989. Greenbelt, MD: Goddardovo centrum kosmických letů v NASA. Dokument SEL-89-007.
- Kovitz, Benjamin, L. (1998): *Practical Software Requirements: A Manual of Content and Style*, Manning Publications Company.
- Kreitzberg, C. B. a B. Shneiderman (1972): *The Elements of Fortran Style*. New York, NY: Harcourt Brace Jovanovich.
- Kruchten, Philippe B. „The 4+1 View Model of Architecture,“ *IEEE Software*, pages 42–50, listopad 1995.
- Kruchten, Philippe (2000): *The Rational Unified Process: An Introduction*, 2. vydání, Reading, MA: Addison-Wesley.
- Kuhn, Thomas S. (1996): *The Structure of Scientific Revolutions*, 3. vydání Chicago: University of Chicago Press.
- Lammers, Susan (1986): *Programmers at Work*. Redmond, WA: Microsoft Press.
- Lampson, Butler (1984): „Hints for Computer System Design,“ *IEEE Software* 1, č. 1 (leden): 11–28.
- Larman, Craig a Rhett Guthrie (2000): *Java 2 Performance and Idiom Guide*. Englewood Cliffs, NJ: Prentice Hall.
- Larman, Craig (2001): *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and the Unified Process*, 2. vydání Englewood Cliffs, NJ: Prentice Hall.
- Larman, Craig (2004): *Agile and Iterative Development: A Manager's Guide*. Boston, MA: Addison-Wesley, 2004.
- Lauesen, Soren. *Software Requirements: Styles and Techniques*. Boston, MA: Addison-Wesley, 2002.
- Laurel, Brenda, ed. (1990): *The Art of Human-Computer Interface Design*. Reading, MA: Addison-Wesley.
- Ledgard, Henry F., with John Tauer (1987a): *C With Excellence: Programming Proverbs*. Indianapolis: Hayden Books.
- Ledgard, Henry F., with John Tauer (1987b): *Professional Software, vol. 2, Programming Practice*. Indianapolis: Hayden Books.
- Ledgard, Henry a Michael Marcotty (1986): *The Programming Language Landscape: Syntax, Semantics a Implementation*, 2. vydání Chicago: Science Research Associates.
- Ledgard, Henry (1985): „Programmers: The Amateur vs. the Professional,“ *Abacus* 2, č. 4 (Summer): 29–35.
- Leffingwell, Dean (1997): „Calculating the Return on Investment from More Effective Requirements Management,“ *American Programmer*, 10(4):13–16.
- Lewis, Daniel W. (1979): „A Review of Approaches to Teaching Fortran,“ *IEEE Transactions on Education*, E-22, č. 1: 23–25.
- Lewis, William E. (2000): *Software Testing and Continuous Quality Improvement*, 2. vydání Auerbach Publishing.
- Lieberherr, Karl J. a Ian Holland (1989): „Assuring Good Style for Object-Oriented Programs,“ *IEEE Software*, září 1989, str. 38f.
- Lientz, B. P. a E. B. Swanson (1980): *Software Maintenance Management*. Reading, MA: Addison-Wesley.
- Lind, Randy K. a K. Vairavan (1989): „An Experimental Investigation of Software Metrics and Their Relationship to Software Development Effort,“ *IEEE*

- Transactions on Software Engineering SE-15, č. 5 (květen): 649–53.
- Linger, Richard C., Harlan D. Mills a Bernard I. Witt (1979): *Structured Programming: Theory and Practice*. Reading, MA: Addison-Wesley.
- Linn, Marcia C. a Michael J. Clancy (1992): „The Case for Case Studies of Programming Problems,” Communications of the ACM 35, č. 3 (březen): 121–32.
- Liskov, Barbara a Stephen Zilles (1974): „Programming with Abstract Data Types,” ACM Sigplan Notices 9, č. 4: 50–59.
- Liskov, Barbara. „Data Abstraction and Hierarchy,” ACM SIGPLAN Notices, květen 1988.
- Littman, David C. a kolektiv (1986): „Mental Models and Software Maintenance,” In Soloway and Iyengar 1986: 80–98.
- Longstreet, David H., ed. (1990): *Software Maintenance and Computers*. Los Alamitos, CA: IEEE Computer Society Press.
- Loy, Patrick H (1990): „A Comparison of Object-Oriented and Structured Development Methods,” Software Engineering Notes 15, č. 1 (leden): 44–48.
- Mackinnon, Tim, Steve Freeman a Philip Craig (2000): „Endo-Testing: Unit Testing with Mock Objects,” eXtreme Programming and Flexible Processes Software Engineering - XP2000 Conference.
- Maguire, Steve (1993): *Writing Solid Code*. Redmond, WA: Microsoft Press.
- Mannino, P. (1987): „A Presentation and Comparison of Four Information System Development Methodologies,” Software Engineering Notes 12, č. 2 (duben): 26–29.
- Manzo, John (2002): „Odyssey and Other Code Science Success Stories,” Crosstalk, říjen 2002.
- Marca, David (1981): „Some Pascal Style Guidelines,” ACM Sigplan Notices 16, č. 4 (duben): 70–80.
- březen, Steve (1999): „Learning from Pathfinder's Bumpy Start,” Software Testing and Quality Engineering, září/říjen 1999, str. 10f.
- Marcotty, Michael (1991): *Software Implementation*. New York, NY: Prentice Hall.
- Martin, Robert C. (2003): *Agile Software Development: Principles, Patterns a Practices*. Upper Saddle River, NJ: Pearson Education.
- McCabe, Tom (1976): „A Complexity Measure,” IEEE Transactions on Software Engineering, SE-2, č. 4 (prosinec): 308–20.
- McCarthy, Jim (1995): *Dynamics of Software Development*. Redmond, WA: Microsoft Press.
- McConnell, Steve (1996): *Rapid Development*. Redmond, WA: Microsoft Press.
- McConnell, Steve (1997a): „The Programmer Writing,” IEEE Software, July/srpen 1997.
- McConnell, Steve (1997b): „Achieving Leaner Software,” IEEE Software, listopad/prosinec 1997.
- McConnell, Steve (1998a): *Software Project Survival Guide*. Redmond, WA: Microsoft Press.
- McConnell, Steve (1998b): „Why You Should Use Routines, Routinely,” IEEE Software, Vol. 15, č. 4, July/srpen 1998.
- McConnell, Steve (1999): „Brooks Law Repealed?,” IEEE Software, listopad/prosinec 1999.
- McConnell, Steve (2004): *Professional Software Development*. Boston, MA: Addison-Wesley.
- McCue, Gerald M. (1978): „IBM's Santa Teresa Laboratory – Architectural Design for Program Development,” IBM Systems Journal 17, č. 1:4–25.
- McGarry, Frank a Rose Pajerski (1990): „Towards Understanding Software – 15 Years in the SEL,” zápis z 15. výročního semináře na téma softwarového inženýrství, listopad 28–29, 1990. Greenbelt, MD: Goddardovo centrum kosmických letů v NASA. Dokument SEL-90-006.
- McGarry, Frank, Sharon Waligora a Tim McDermott (1989): „Experiences in the Software Engineering Laboratory (SEL) Applying Software Measurement,” zápis z 14. výročního semináře na téma softwarového inženýrství, listopad 29, 1989. Greenbelt, MD: Goddardovo centrum kosmických letů v NASA. Dokument SEL-89-007.
- McGarry, John, a kolektiv (2001): *Practical Software Measurement: Objective Information for Decision Makers*. Boston, MA: Addison-Wesley.
- McKeithen, Katherine B. a kolektiv (1981): „Knowledge Organization and Skill Differences in Computer Programmers,” Cognitive Psychology 13:307–25.
- Metzger, Philip W. a John Boddie (1996): *Managing a Programming Project: Processes and People*, 3. vydání Englewood Cliffs, NJ: Prentice Hall, 1996.
- Meyer, Bertrand (1997): *Object-Oriented Software Construction*, 2. vyd., New York, NY: Prentice Hall.
- Meyers, Scott (1996): *More Effective C++: 35 New Ways to Improve Your Programs and Designs*. Reading, MA: Addison-Wesley.
- Meyers, Scott (1998): *Effective C++: 50 Specific Ways to Improve Your Programs and Designs*, 2. vydání, Reading, MA: Addison-Wesley.

- Miaria, Richard J. a kolektiv (1983): „*Program Indentation and Comprehensibility*,“ Communications of the ACM 26, č. 11 (listopad): 861–67.
- Michalewicz, Zbigniew a David B. Fogel (2000): *How to Solve It: Modern Heuristics*. Berlin: Springer-Verlag.
- Miller, G. A. (1956): „*The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information*,“ The Psychological Review 63, č. 2 (březen): 81–97.
- Mills, Harlan D. (1983): *Software Productivity*. Boston, MA: Little, Brown.
- Mills, Harlan D. (1986): „*Structured Programming: Retrospect and Prospect*,“ IEEE Software, listopad, 58–66.
- Mills, Harlan D. a Richard C. Linger (1986): „*Data Structured Programming: Program Design Without Arrays and Pointers*,“ IEEE Transactions on Software Engineering SE-12, č. 2 (únor): 192–97.
- Mills, Harlan D., Michael Dyer a Richard C. Linger (1987): „*Cleanroom Software Engineering*,“ IEEE Software, září, 19–25.
- Misfeldt, Trevor, Greg Bumgardner a Andrew Gray (2004): *The Elements of C++ Style*. Cambridge University Press.
- Mitchell, Jeffrey, Joseph Urban a Robert McDonald (1987): „*The Effect of Abstract Data Types on Program Development*,“ IEEE Computer 20, č. 9 (září): 85–88.
- Mody, R. P. (1991): „*C in Education and Software Engineering*,“ SIGCSE Bulletin 23, č. 3 (září): 45–56.
- Moore, Dave (1992): *Private communication*. Moore, James W. (1997): *Software Engineering Standards: A User's Road Map*. Los Alamitos, CA: IEEE Computer Society Press.
- Morales, Alexandra Weber (2003): „*The Consummate Coach: Watts Humphrey, Father of Cmm and Author of Winning with Software, Explains How to Get Better at What You Do*,“ SD Show Daily, září 16, 2003.
- Myers, Glenford J. (1976): *Software Reliability*. New York, NY: John Wiley & Sons.
- Myers, Glenford J. (1978a): *Composite/Structural Design*. New York, NY: Van Nostrand Reinhold.
- Myers, Glenford J. (1978b): „*A Controlled Experiment in Program Testing and Code Walk-throughs/Inspections*,“ Communications of the ACM 21, č. 9 (září): 760–68.
- Myers, Glenford J. (1979): *The Art of Software Testing*. New York, NY: John Wiley & Sons.
- Myers, Ware (1992): „*Good Software Practices Pay Off— Or Do They?*,“ IEEE Software, březen, 96–97.
- Naisbitt, John (1982): *Megatrends*. New York, NY: Warner Books.
- NASA Software Engineering Laboratory, 1994. Software Measurement Guidebook, červen 1995, NASA-GB-001-94. Ke stažení na adrese <http://sel.gsfc.nasa.gov/web-site/documents/online-doc/94-102.pdf>.
- NCES 2002. National Center for Education Statistics, 2001 Digest of Educational Statistics, Dokument č. NCES 2002130, duben 2002.
- Nevison, John M. (1978): *The Little Book of BASIC Style*. Reading, MA: Addison-Wesley.
- Newcomer, Joseph M. (2000): „*Optimization: Your Worst Enemy*,“ květen 2000, www.flounder.com/optimization.htm.
- Norcio, A. F. (1982): „*Indentation, Documentation and Programmer Comprehension*,“ zápis: lidský faktor v počítačových systémech, březen 15–17, 1982, Gaithersburg, MD: 118–20.
- Norman, Donald A. (1988): *The Psychology of Everyday Things*. New York, NY: Basic Books. (Vydáno rovněž jako brožovaný výtisk *The Design of Everyday Things*. New York, NY: Doubleday, 1990.)
- Oman, Paul a Shari Lawrence Pfleeger, eds. (1996): *Applying Software Metrics*. Los Alamitos, CA: IEEE Computer Society Press.
- Oman, Paul W. a Curtis R. Cook (1990a): „*The Book Paradigm for Improved Maintenance*,“ IEEE Software, leden, 39–45.
- Oman, Paul W. a Curtis R. Cook (1990b): „*Typographic Style Is More Than Cosmetic*,“ Communications of the ACM 33, č. 5 (květen): 506–20.
- Ostrand, Thomas J. a Elaine J. Weyuker (1984): „*Collecting and Categorizing Software Error Data in an Industrial Environment*,“ Journal of Systems and Software 4, č. 4 (listopad): 289–300.
- Page-Jones, Meilir (2000): *Fundamentals of Object-Oriented Design in UML*. Boston, MA: Addison-Wesley.
- Page-Jones, Meilir (1988): *The Practical Guide to Structured Systems Design*. Englewood Cliffs, NJ: Yourdon Press.
- Parikh, G. a N. Zvegintzov, eds. (1983): *Tutorial on Software Maintenance*. Los Alamitos, CA: IEEE Computer Society Press.
- Parikh, Girish (1986): *Handbook of Software Maintenance*. New York, NY: John Wiley & Sons.
- David L. Parnas: *On the Criteria to Be Used in Decomposing Systems into Modules* (O kritériích, která bychom měli používat při rozkladu systému na modu-

- ly), Communications of the ACM 5, č. 12 (prosinec 1972): 1053–58.
- Parnas, David L. (1976): „On the Design and Development of Program Families,” IEEE Transactions on Software Engineering SE-2, 1 (březen): 1–9.
- Parnas, David L. (1979): „Designing Software for Ease of Extension and Contraction,” IEEE Transactions on Software Engineering SE-5, č. 2 (březen): 128–38.
- Parnas, David L. (1999): *ACM Fellow Profile: David Lorge Parnas*. ACM Software Engineering Notes, květen 1999, 10–14.
- Parnas, David L. a Paul C. Clements (1986): „A Rational Design Process: How and Why to Fake It,” IEEE Transactions on Software Engineering SE-12, č. 2 (únor): 251–57.
- Parnas, David L., Paul C. Clements a D. M. Weiss (1985): „The Modular Structure of Complex Systems,” IEEE Transactions on Software Engineering SE-11, č. 3 (březen): 259–66.
- Perrott, Pamela (2004): *Private communication*. Peters, L. J. a L. L. Tristr (1976): „Is Software Design Wicked,” Datamation, Vol. 22, č. 5 (květen 1976), 127–136.
- Peters, Lawrence J. (1981): *Handbook of Software Design: Methods and Techniques*. New York, NY: Yourdon Press.
- Peters, Lawrence J. a Leonard L. Tristr (1977): „Comparing Software Design Methodologies,” Datamation, listopad, 89–94.
- Peters, Tom (1987): *Thriving on Chaos: Handbook for a Management Revolution*. New York, NY: Knopf.
- Petroski, Henry (1994): *Design Paradigms: Case Histories of Error and Judgment in Engineering*. Cambridge, U.K.: Cambridge University Press.
- Pietrasanta, Alfred M. (1990): „Alfred M. Pietrasanta on Improving the Software Process,” Software Engineering: Tools, Techniques, Practices 1, č. 1 (květen/červen): 29–34.
- Pietrasanta, Alfred M. (1991a): „A Strategy for Software Process Improvement,” Ninth Annual Pacific Northwest Software Quality Conference, říjen 7–8, 1991. Oregon Convention Center, Portland, OR.
- Pietrasanta, Alfred M. (1991b): „Implementing Software Engineering in IBM,” Keynote address. Ninth Annual Pacific Northwest Software Quality Conference, říjen 7–8, 1991. Oregon Convention Center, Portland, OR.
- Pigoski, Thomas M. (1997): *Practical Software Maintenance*. New York, NY: John Wiley & Sons.
- Pirsig, Robert M. (1974): *Zen and the Art of Motorcycle Maintenance: An Inquiry into Values*. William Morrow.
- Plauser, P. J. (1988): „A Designer's Bibliography,” Computer Language, July, 17–22.
- Plauser, P. J. (1993): *Programming on Purpose: Essays on Software Design*. New York, NY: Prentice Hall.
- Plum, Thomas (1984): *C Programming Guidelines*. Cardiff, NJ: Plum Hall.
- Polya, G. (1957): *How to Solve It: A New Aspect of Mathematical Method*, 2. vydání Princeton, NJ: Princeton University Press.
- Post, Ed. 1983. „Real Programmers Don't Use Pascal,” Datamation, July 1983, 263–265.
- Prechelt, Lutz (2000): „An Empirical Comparison of Seven Programming Languages,” IEEE Computer, říjen 2000, 23–29.
- Pressman, Roger S. (1987): *Software Engineering: A Practitioner's Approach*. New York, NY: McGraw-Hill.
- Pressman, Roger S. (1988): *Making Software Engineering Happen: A Guide for Instituting the Technology*. Englewood Cliffs, NJ: Prentice Hall.
- Putnam, Lawrence H. (2000): „Familiar Metric Management – Effort, Development Time a Defects Interact,” Downloadable from www.qsm.com.
- Putnam, Lawrence H. a Ware Myers (1992): *Measures for Excellence: Reliable Software On Time, Within Budget*. Englewood Cliffs, NJ: Yourdon Press, 1992.
- Putnam, Lawrence H. a Ware Myers (1997): *Industrial Strength Software: Effective Management Using Measurement*. Washington, DC: IEEE Computer Society Press.
- Putnam, Lawrence H. a Ware Myers (2000): „What We Have Learned,” Ke stažení z adresy www.qsm.com, červen 2000.
- Raghavan, Sridhar A. a Donald R. Chand (1989): „Diffusing Software-Engineering Methods,” IEEE Software, July, 81–90.
- Ramsey, H. Rudy, Michael E. Atwood a James R. Van Doren (1983): „Flowcharts Versus Program Design Languages: An Experimental Comparison,” Communications of the ACM 26, č. 6 (červen): 445–49.
- Ratliff, Wayne (1987): Interview in Solution System.
- Raymond, E. S. (2000): „The Cathedral and the Bazaar,” www.catb.org/~esr/writings/cathedral-bazaar.

- Raymond, Eric S. (2004): *The Art of Unix Programming*. Boston, MA: Addison-Wesley.
- Rees, Michael J. (1982): „Automatic Assessment Aids for Pascal Programs“, ACM Sigplan Notices 17, č. 10 (říjen): 33–42.
- Reifer, Donald (2002): „How to Get the Most Out of Extreme Programming/Agile Methods“, zápis, XP/Agile Universe 2002. New York, NY: Springer; 185–196.
- Reingold, Edward M. a Wilfred J. Hansen (1983): *Data Structures*. Boston, MA: Little, Brown.
- Rettig, Marc (1991): „Testing Made Palatable“, Communications of the ACM 34, č. 5 (květen): 25–29.
- Riel, Arthur J. (1996): *Object-Oriented Design Heuristics*. Reading, MA: Addison-Wesley.
- Rittel, Horst a Melvin Webber (1973): „Dilemmas in a General Theory of Planning“, Policy Sciences 4:155–69.
- Robertson, Suzanne a James Robertson (1999): *Mastering the Requirements Process*. Reading, MA: Addison-Wesley.
- Rogers, Everett M. (1995): *Diffusion of Innovations*, 4. vydání New York, NY: The Free Press.
- Rombach, H. Dieter (1990): „Design Measurements: Some Lessons Learned“, IEEE Software, březien, 17–25.
- Rubin, Frank (1987): „‘GOTO Considered Harmful’ Considered Harmful“, Letter to the editor. Communications of the ACM 30, č. 3 (březien): 195–96. Následují články v 30, č. 5 (květen 1987): 351–55; 30, č. 6 (červen 1987): 475–78; 30, č. 7 (July 1987): 632–34; 30, č. 8 (srpen 1987): 659–62; 30, č. 12 (prosinec 1987): 997, 1085.
- Sackman, H., W. J. Erikson a E. E. Grant (1968): „Exploratory Experimental Studies Comparing Online and Offline Programming Performance“, Communications of the ACM 11, č. 1 (leden): 3–11.
- Schneider, G. Michael, Johnny Martin a W. T. Tsai (1992): „An Experimental Study of Fault Detection in User Requirements Documents“, ACM Transactions on Software Engineering and Methodology, svazek 1, č. 2, 188–204.
- Schulmeyer, G. Gordon (1990): *Zero Defect Software*. New York, NY: McGraw-Hill.
- Sedgewick, Robert (1997): *Algorithms in C, Parts 1–4*, 3. vydání, Boston, MA: Addison-Wesley.
- Sedgewick, Robert (2001): *Algorithms in C, Part 5*, 3. vydání, Boston, MA: Addison-Wesley.
- Sedgewick, Robert (1998): *Algorithms in C++, Parts 1–4*, 3. vydání, Boston, MA: Addison-Wesley.
- Sedgewick, Robert (2002): *Algorithms in C++, Part 5*, 3. vydání, Boston, MA: Addison-Wesley.
- Sedgewick, Robert (2002): *Algorithms in Java, Parts 1–4*, 3. vydání, Boston, MA: Addison-Wesley.
- Sedgewick, Robert (2003): *Algorithms in Java, Part 5*, 3. vydání, Boston, MA: Addison-Wesley.
- SEI 1995. The Capability Maturity Model: *Guidelines for Improving the Software Process*, Software Engineering Institute, Reading, MA: Addison-Wesley, 1995.
- SEI, 2003. „Process Maturity Profile: Software CMM®, CBA IPI and SPA Appraisal Results: 2002 Year End Update“, Software Engineering Institute, duben 2003.
- Selby, Richard W. a Victor R. Basili (1991): „Analyzing Error-Prone System Structure“, IEEE Transactions on Software Engineering SE-17, č. 2 (únor): 141–52.
- SEN 1990. „Subsection on Telephone Systems“, Software Engineering Notes, duben 1990, 11–14.
- Shalloway, Alan a James R. Trott (2002): *Design Patterns Explained*. Boston, MA: Addison-Wesley.
- Sheil, B. A. (1981): „The Psychological Study of Programming“, Computing Surveys 13, č. 1 (březien): 101–20.
- Shen, Vincent Y. a kolektiv (1985): „Identifying Error-Prone Software – An Empirical Study“, IEEE Transactions on Software Engineering SE-11, č. 4 (duben): 317–24.
- Sheppard, S. B. a kolektiv (1978): „Predicting Programmers’ Ability to Modify Software“, TR 78-388100-3, General Electric Company, květen.
- Sheppard, S. B. a kolektiv (1979): „Modern Coding Practices and Programmer Performance“, IEEE Computer 12, č. 12 (prosinec): 41–49.
- Shepperd, M. a D. Ince (1989): „Metrics, Outlier Analysis and the Software Design Process“, Information and Software Technology 31, č. 2 (březien): 91–98.
- Shirazi, Jack (2000): *Java Performance Tuning*. Sebastopol, CA: O’Reilly & Associates.
- Shlaer, Sally a Stephen J. Mellor (1988): *Object Oriented Systems Analysis – Modeling the World in Data*. Englewood Cliffs, NJ: Prentice Hall.
- Shneiderman, Ben a Richard květener (1979): „Syntactic/Semantic Interactions in Programmer Behavior: A Model and Experimental Results“, International Journal of Computer and Information Sciences 8, č. 3: 219–38.
- Shneiderman, Ben (1976): „Exploratory Experiments in Programmer Behavior“, International Journal of Computing and Information Science 5:123–43.

- Shneiderman, Ben (1980): *Software Psychology: Human Factors in Computer and Information Systems*. Cambridge, MA: Winthrop.
- Shneiderman, Ben (1987): *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Reading, MA: Addison-Wesley.
- Shull a kolektiv (2002): „*What We Have Learned About Fighting Defects*,“ *zápis, Metrics 2002*. IEEE; 249–258.
- Simon, Herbert (1996): *The Sciences of the Artificial*, 3. vydání, Cambridge, MA: MIT Press.
- Simon, Herbert. *The Shape of Automation for Men and Management*. Harper and Row, 1965.
- Simonyi, Charles a Martin Heller (1991): „*The Hungarian Revolution*,“ *BYTE*, srpen, 131–38.
- Smith, Connie U. a Lloyd G. Williams (2002): *Performance Solutions: A Practical Guide to Creating Responsive, Scalable Software*. Boston, MA: Addison-Wesley.
- Software Productivity Consortium (1989): *Ada Quality and Style: Guidelines for Professional Programmers*. New York, NY: Van Nostrand Reinhold.
- Soloway, Elliot a Kate Ehrlich (1984): „*Empirical Studies of Programming Knowledge*,“ *IEEE Transactions on Software Engineering SE-10*, č. 5 (září): 595–609.
- Soloway, Elliot a Sitharama Iyengar, eds (1986): *Empirical Studies of Programmers*. Norwood, NJ: Ablex.
- Soloway, Elliot, Jeffrey Bonar a Kate Ehrlich (1983): „*Cognitive Strategies and Looping Constructs: An Empirical Study*,“ *Communications of the ACM 26*, č. 11 (listopad): 853–60.
- Solution Systems (1987): *World-Class Programmers' Editing Techniques: Interviews with Seven Programmers*. South Weymouth, MA: Solution Systems.
- Sommerville, Ian (1989): *Software Engineering*, 3. vydání, Reading, MA: Addison-Wesley.
- Spier, Michael J. (1976): „*Software Malpractice – A Distasteful Experience*,“ *Software – Practice and Experience 6*: 293–99.
- Spinellis, Diomidis (2003): *Code Reading: The Open Source Perspective*. Boston, MA: Addison-Wesley.
- SPMN. (1998): *Little Book of Configuration Management*. Arlington, VA; Software Program Managers Network. Starr, Daniel (2003): „*What Supports the Roof?*,“ *Software Development*. July 2003, 38–41.
- Stephens, Matt (2003): „*Emergent Design vs. Early Prototyping*,“ květen 26, 2003, www.softwarereality.com/design/early_prototyping.jsp
- Stevens, Scott M. (1989): „*Intelligent Interactive Video Simulation of a Code Inspection*,“ *Communications of the ACM 32*, č. 7 (July): 832–43.
- Stevens, W., G. Myers a L. Constantine (1974): „*Structured Design*,“ *IBM Systems Journal 13*, č. 2 (květen): 115–39.
- Stevens, Wayne (1981): *Using Structured Design*. New York, NY: John Wiley & Sons.
- Stroustrup, Bjarne (1997): *The C++ Programming Language*, 3. vydání, Reading, MA: Addison-Wesley.
- Strunk, William a E. B. White (2000): *Elements of Style*, 4. vydání, Pearson.
- Sun Microsystems, Inc. (2000): „*How to Write Doc Comments for the Javadoc Tool*,“ 2000. Ke stažení na adrese <http://java.sun.com/j2se/javadoc/writingdoccomments>.
- Sutter, Herb (2000): *Exceptional C++: 47 Engineering Puzzles, Programming Problems a Solutions*. Boston, MA: Addison-Wesley.
- Tackett, Buford D., III a Buddy Van Doren (1999): „*Process Control for Error Free Software: A Software Success Story*,“ *IEEE Software*, květen 1999.
- Tenner, Edward (1997): *Why Things Bite Back: Technology and the Revenge of Unintended Consequences*. Vintage Books.
- Tenny, Ted (1988): „*Program Readability: Procedures versus Comments*,“ *IEEE Transactions on Software Engineering SE-14*, č. 9 (září): 1271–79.
- Thayer, Richard H., ed (1990): *Tutorial: Software Engineering Project Management*. Los Alamitos, CA: IEEE Computer Society Press.
- Thimbleby, Harold (1988): „*Delaying Commitment*,“ *IEEE Software*, květen, 78–86.
- Thomas, Dave a Andy Hunt (2002): „*Mock Objects*,“ *IEEE Software*, květen/červen 2002.
- Thomas, Edward J. a Paul W. Oman (1990): „*A Bibliography of Programming Style*,“ *ACM Sigplan Notices 25*, č. 2 (únor): 7–16.
- Thomas, Richard A. (1984): „*Using Comments to Aid Program Maintenance*,“ *BYTE*, květen, 415–22.
- Tripp, Leonard L., William F. Struck a Bryan K. Pflug (1991): „*The Application of Multiple Team Inspections on a Safety-Critical Software Standard*,“ *zápis ze 4. semináře na téma uplatnění standardů softwarového inženýrství*, Los Alamitos, CA: IEEE Computer Society Press.

- U.S. Department of Labor (1990): „*The 1990-91 Job Outlook in Brief*,“ Occupational Outlook Quarterly, Spring. U.S. Government Printing Office. Dokument 1990-282-086/20007.
- Valett, J. a F. E. McGarry (1989): „*A Summary of Software Measurement Experiences in the Software Engineering Laboratory*,“ Journal of Systems and Software 9, č. 2 (únor): 137-48.
- Van Genuchten, Michiel (1991): „*Why Is Software Late? An Empirical Study of Reasons for Delay in Software Development*,“ IEEE Transactions on Software Engineering SE-17, č. 6 (červen): 582-90.
- Van Tassel, Dennie (1978): *Program Style, Design, Efficiency, Debugging a Testing*, 2. vydání, Englewood Cliffs, NJ: Prentice Hall.
- Vaughn-Nichols, Steven (2003): „*Building Better Software with Better Tools*,“ IEEE Computer, září 2003, 12-14.
- Vermeulen, Allan, a kolektiv (2000): *The Elements of Java Style*. Cambridge University Press.
- Vessey, Iris, Sirkka L. Jarvenpaa a Noam Tractinsky (1992): „*Evaluation of Vendor Products: CASE Tools as Methodological Companions*,“ Communications of the ACM 35, č. 4 (duben): 91-105.
- Vessey, Iris (1986): „*Expertise in Debugging Computer Programs: An Analysis of the Content of Verbal Protocols*,“ IEEE Transactions on Systems, Man a Cybernetics SMC-16, č. 5 (září/říjen): 621-37.
- Votta, Lawrence G. a kolektiv (1991): „*Investigating the Application of Capture-Recapture Techniques to Requirements and Design Reviews*,“ zápis z 16. výročního semináře na téma softwarového inženýrství, prosinec 4-5, 1991. Greenbelt, MD: Goddardovo centrum kosmických letů v NASA. Dokument SEL-91-006.
- Walston, C. E. a C. P. Felix (1977): „*A Method of Programming Measurement and Estimation*,“ IBM Systems Journal 16, č. 1: 54-73.
- Ward, Robert (1989): *A Programmer's Introduction to Debugging*. C. Lawrence, KS: R & D Publications.
- Ward, William T. (1989): „*Software Defect Prevention Using McCabe's Complexity Metric*,“ Hewlett-Packard Journal, duben, 64-68.
- Webster, Dallas E. (1988): „*Mapping the Design Information Representation Terrain*,“ IEEE Computer, prosinec, 8-23.
- Weeks, Kevin (1992): „*Is Your Code Done Yet?*,“ Computer Language, duben, 63-72.
- Weiland, Richard J. (1983): *The Programmer's Craft: Program Construction, Computer Architecture a Data Management*. Reston, VA: Reston Publishing.
- Weinberg, Gerald M. (1983): „*Kill That Code!*,“ Info-systems, srpen, 48-49.
- Weinberg, Gerald M. (1998): *The Psychology of Computer Programming: Silver Anniversary Edition*. New York, NY: Dorset House.
- Weinberg, Gerald M. a Edward L. Schulman (1974): „*Goals and Performance in Computer Programming*,“ Human Factors 16, č. 1 (únor): 70-77.
- Weinberg, Gerald (1988): *Rethinking Systems Analysis and Design*. New York, NY: Dorset House.
- Weisfeld, Matt (2004): *The Object-Oriented Thought Process*, 2. vydání, SAMS, 2004.
- Weiss, David M. (1975): „*Evaluating Software Development by Error Analysis: The Data from the Architecture Research Facility*,“ Journal of Systems and Software 1, č. 2 (červen): 57-70.
- Weiss, Eric A. (1972): „*Review of The Psychology of Computer Programming, by Gerald M. Weinberg*,“ ACM Computing Reviews 13, č. 4 (duben): 175-76.
- Wheeler, David, Bill Brykczynski a Reginald Meeson (1996): *Software Inspection: An Industry Best Practice*. Los Alamitos, CA: IEEE Computer Society Press.
- Whittaker, James A. (2000): „*What Is Software Testing? And Why Is It So Hard?*,“ IEEE Software, leden 2000, 70-79.
- Whittaker, James A. (2002): *How to Break Software: A Practical Guide to Testing*. Boston, MA: Addison-Wesley.
- Whorf, Benjamin (1956): *Language, Thought and Reality*. Cambridge, MA: MIT Press.
- Wiegers, Karl (2002): *Peer Reviews in Software: A Practical Guide*. Boston, MA: Addison-Wesley.
- Wiegers, Karl (2003): *Software Requirements*, 2. vydání Redmond, WA: Microsoft Press.
- Williams, Laurie a Robert Kessler (2002): *Pair Programming Illuminated*. Boston, MA: Addison-Wesley.
- Willis, Ron R. a kolektiv (1998): „*Hughes Aircraft's Widespread Deployment of a Continuously Improving Software Process*,“ Software Engineering Institute/Carnegie Mellon University, CMU/SEI-98-TR-006, květen 1998.
- Wilson, Steve a Jeff Kesselman (2000): *Java Platform Performance: Strategies and Tactics*. Boston, MA: Addison-Wesley.
- Wirth, Niklaus (1995): „*A Plea for Lean Software*,“ IEEE Computer, únor 1995.

Wirth, Niklaus (1971): „*Program Development by Stepwise Refinement*,“ Communications of the ACM 14, č. 4 (duben): 221–27.

Wirth, Niklaus (1986): *Algorithms and Data Structures*. Englewood Cliffs, NJ: Prentice Hall.

Woodcock, Jim a Martin Loomes (1988): *Software Engineering Mathematics*. Reading, MA: Addison-Wesley.

Woodfield, S. N., H. E. Dunsmore a V. Y. Shen (1981): „*The Effect of Modularization and Comments on Program Comprehension*,“ zápis z 5. mezinárodní konference na téma softwarového inženýrství, březen 1981, 215–23.

Wulf, W. A. (1972): „*A Case Against the GOTO*,“ zápis z 25. národní konference ACM, srpen 1972, 791–97.

Youngs, Edward A. (1974): „*Human Errors in Programming*,“ International Journal of Man-Machine Studies 6:361–76.

Yourdon, Edward a Larry L. Constantine (1979): *Structured Design: Fundamentals of a Discipline of Computer Program and Systems Design*. Englewood Cliffs, NJ: Yourdon Press.

Yourdon, Edward, ed (1979): *Classics in Software Engineering*. Englewood Cliffs, NJ: Yourdon Press.

Yourdon, Edward, ed (1982): *Writings of the Revolution: Selected Readings on Software Engineering*. New York, NY: Yourdon Press.

Yourdon, Edward (1986a): *Managing the Structured Techniques: Strategies for Software Development in the 1990s*, 3. vydání New York, NY: Yourdon Press.

Yourdon, Edward (1986b): *Nations at Risk*. New York, NY: Yourdon Press.

Yourdon, Edward (1988): „*The 63 Greatest Software Books*,“ American Programmer, září.

Yourdon, Edward (1989a): *Modern Structured Analysis*. New York, NY: Yourdon Press.

Yourdon, Edward (1989b): *Structured Walk-Throughs*, 4. vydání, New York, NY: Yourdon Press.

Yourdon, Edward (1992): *Decline & Fall of the American Programmer*. Englewood Cliffs, NJ: Yourdon Press.

Zachary, Pascal (1994): *Showstopper!* The Free Press.

Zahniser, Richard A. (1992): „*A Massively Parallel Software Development Approach*,“ American Programmer, leden, 34–41.

Rejstřík