

Bibliography

(Aho et al, 2007) Alfred V Aho, Monica S Lam, Ravi Sethi, and Jeffrey D Ullman. *Compilers – Principles, Techniques, & Tools (Second Edition)*. Pearson Education, Addison Wesley, 2007.

(Aho, Johnson and Ullman, 1975) Alfred V Aho, Johnson, and Jeffrey D Ullman. “Deterministic parsing of ambiguous grammars.” *Communications of the ACM* 18(8), 1975, pp. 441-452.

(Aho and Ullman, 1972) Alfred V Aho and Jeffrey D Ullman. *The Theory of Parsing, Translation and Compiling*. Prentice-Hall, 1972.

(Apache Tomcat, 2006) “Apache Tomcat 6.0 Release Notes, 2006.” Available on the WWW at <http://tomcat.apache.org/tomcat-6.0-doc/RELEASE-NOTES.txt>.

(Appel, 2002) Andrew W, Appel with Jens Palsberg. *Modern Compiler Implementation in Java (Second Edition)*. Cambridge University Press, New York, NY, 2004.

(Aycock, 2003) John Aycock. “A Brief History of Just-In-Time.” *ACM Computing Surveys* vol.35(2). Association for Computing Machinery, 2003.

(Beck, 1999) Kent Beck. *Extreme Programming Explained: Embrace Change*. Addison-Wesley Professional, October 5, 1999.

(Bison, 2008) Charles Donnelly and Richard Stallman. *Bison, the YACC-compatible parser generator*. Version 2.3, 2006.

(Bracha, 2004) Gilad Bracha. *Generics in the Java Programming Language*. Oracle, 2004. Available at <http://java.sun.com/j2se/1.5/pdf/generics-tutorial.pdf>.

(Briggs et al, 1994) P. Briggs, K. D. Cooper, and L. Torczon. “Improvements to graph coloring register allocation.” *ACM Transactions on Programming Languages and Systems* 16(3), pp. 428 – 455. ACM, 1994.

(Bothner, 2003) Per Bothner. “Compiling Java with GCJ.” *Linux Journal* 105, January 2003.

(Chaitin, et al, 1981) Gregory J. Chaitin, Marc A. Auslander, Ashok. K. Chandra, John Cocke, Martin. E. Hopkins, Peter W. Markstein. “Register allocation via coloring.” *Computer Languages*, Volume 6 (1981), pp 47-57. Elsevier Ltd, 1981.

(Chaitin, 1982) Gregory J. Chaitin. “Register allocation & spilling via graph coloring.” *Proceedings of the 1982 SIGPLAN symposium on Compiler construction*, pp 98-101. ACM Press, 1982.

(Clayberg and Rubel, 2008) Eric Clayberg and Dan Rubel. *Eclipse Plug-ins (Third Edition)*. Addison-Wesley Professional, 2008.

(CLI Spec, 2006) *Common Language Infrastructure - Partition I: Concepts and Architecture, With Added Microsoft Specific Implementation Notes*, 2006. Available on the WWW at <http://msdn.microsoft.com/en-us/netframework/aa569283>.

(CLR, 2011) MSDN Library. *Common Language Runtime (CLR)* on the WWW at <http://msdn.microsoft.com/en-us/library/8bs2ecf4.aspx>.

(Cooper and Dasgupta, 2006) Keith D. Cooper and Anshuman Dasgupta. “Tailoring Graph-coloring Register Allocation For Runtime Compilation.” *Proceedings of the 2006 International Symposium on Code Generation and Optimization*, IEEE Computer Society, 2006.

(Copeland, 2007) Tom Copeland. *Generating Parsers with JavaCC*. Centennial Books, 2007.

(Corliss and Lewis, 2007) Marc L. Corliss and E. Christopher Lewis. *Bantam Java Runtime System Manual*. Part of the documentation for the Bantum Java compiler project, on the WWW at <http://www.bantamjava.com>, 2007.

(D'Anjou et al, 2004) Jim D'Anjou, Scott Fairbrother, Dan Kehn, John Kellerman, and Pat McCarthy. *Java Developer's Guide to Eclipse (Second Edition)*. Addison-Wesley Professional, 2004.

(DeRemer, 1969) F DeRemer. *Practical Translators for LR(k) Languages*. PhD Dissertation, MIT, 1969.

(Eclipse Documentation) on the WWW at <http://help.eclipse.org/indigo/index.jsp>.

(Eclipse, 2004) John Arthorne. *Project Builders and Natures*. IBM OTI Labs, 2004. Available via <http://www.eclipse.org/articles/Article-Builders/builders.html>.

(Eclipse, 2006) *Eclipse Platform Technical Overview*. IBM Corp., 2006. On the WWW at <http://www.eclipse.org/articles/Whitepaper-Platform-3.1/eclipse-platform-whitepaper.pdf>

(Eclipse, 2011) *JDT Core Component webpage* on the WWW at <http://www.eclipse.org/jdt/core/>.

Bibliography - 2

(Flex, 2008) Flex was originally written by Vern Paxton. Flex home page <http://flex.sourceforge.net/>.

(Freeman et al, 2004) Elisabeth Freeman, Eric Freeman, Bert Bates, Kathy Sierra. *Head First Design Patterns*. O'Reilly Media, 2004.

(Gamma, 1995) Erich Gamma, Richard Helm, Ralph Johnson and John Vlissides. *Design Patterns – Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional, 1995.

(GCJ, 2011) *The GCJ FAQ* on the WWW at <http://gcc.gnu.org/java/faq.html>.

(George and Appel, 1996) L. George and A. W. Appel. "Iterated register coalescing." *ACM Transactions on Programming Languages*, 18(3), 300-324. ACM, 2006.

(GNU, 2008) *GCC Wiki* on the WWW at <http://gcc.gnu.org/wiki> .

(GNU, 2010) *GNU Manual*, Section 3.10, "Options That Control Optimization" on the WWW at <http://gcc.gnu.org/onlinedocs/gcc/Optimize-Options.html>.

(GNU, 2011) *The GNU Compiler for the Java™ Programming Language* home page on the WWW at <http://gcc.gnu.org/java/> .

(Goetz, 2004) Brian Goetz. *Java theory and practice: Dynamic compilation and performance measurement - The perils of benchmarking under dynamic compilation*. IBM Technical Library, 2004. Available on the WWW at <http://www.ibm.com/developerworks/library/j-jtp12214/>.

(Gosling et al, 2005) James Gosling, Bill Joy, Guy Steele, Jr., and Gilad Bracha. *The Java Language Specification (Third Edition)*, Addison-Wesley Professional, June 14, 2005. This book is also published on line on the Sun Developer Network at <http://java.sun.com/docs/books/jls>.

(Gosling, 2000) James Gosling, Bill Joy, Guy Steele, Jr., and Gilad Bracha. *The Java Language Specification (Second Edition)*, Prentice Hall PTR, June 15, 2000. This book is also published online on the Sun Developer Network at http://java.sun.com/docs/books/jls/second_edition/html/j.title.doc.html.

(Gough, 2001) John Gough. *Compiling for the .Net Common Language Runtime*. Prentice-Hall PTR, November 2001.

(Gries, 1971) *Compiler Construction for Digital Computers*. John Wiley and Sons, New York, 1971.

Bibliography - 3

(Hamilton, 2003) Jennifer Hamilton. “Language Integration in the Common Language Runtime.” *ACM Sigplan Notices*, Vol. 38(2), 2003.

(Hopcroft and Ullman, 1969) John E Hopcroft and Jeffrey D Ullman. *Formal Languages and their relation to Automata*. Addison Wesley Publishing Company, 1969.

(JavaCC, 2011) JavaCC™ – the Java Parser Generator. Available on the WWW at <http://javacc.java.net/>.

(Jia, 2003) Xiaoping Jia, *Object-Oriented Software Development Using Java (Second Edition)*. Addison Wesley Publishing Company, 2003.

(JetBrains, 2011) IntelliJ IDEA *Compiling/ Running/ Debugging*. Jet Brains, 2011. Available via <http://www.jetbrains.com/idea/features/compiler.html>.

(Johnson, 1975) Stephen C. Johnson. “Yacc -- yet another compiler compiler,” *Computing Science Technical Report 32*, AT&T Bell Laboratories, Murray Hill, N.J.

(Jones, 1996 and 1999) Richard Jones. *Garbage Collection – algorithms for automatic dynamic memory management*. John Wiley & Sons, Ltd, 1996 and 1999.

(Jones, 2005) M. Tim Jones. “Optimization in GCC.” *Linux Journal*, 131, March 2005.

(Kazi et al, 2000) Iffat H. Kazi, Howard H. Chen, Berdenia Stanley and David J. Lilja. “Techniques for Obtaining High Performance in Java Programs.” *ACM Computing Surveys*, 32(3), 2000, pp. 213–240.

(Knuth, 1965) D.E. Knuth. “On the translation of languages from left to right.” *Information and Control* 8(6), 1965, pp. 607-639.

(Knuth, 1968) D.E. Knuth. “Semantics of Context-free Languages.” *Mathematical Systems Theory*, 2(2). Springer-Verlag, 1968, pp. 127-145.

(Knuth, 1971) Donald Knuth. “Top-Down Syntax Analysis.” *Acta Informatica* 1(2), 1971.

(Knuth, 1971) Donald Knuth. “An Empirical Study of Fortran programs.” *Software: Practice and Experience* 1, 1971 105-133.

(Kommalapati and Christian, 2005) Hanu Kommalapati and Tom Christian. “Drill Into .NET Framework Internals to See How the CLR Creates Runtime Objects.” *MSDN Magazine*, May 2005.

Bibliography - 4

(Larus, 2000-2010) James Larus. *SPIM A MIPS32 Simulator*. On the WWW at <http://spimsimulator.sourceforge.net/>.

(Larus, 2009) James Larus. “Assemblers, Linkers, and the SPIM Simulator.” Appendix B in David A. Patterson and John L. Hennessy, *Computer Organization and Design, Fourth Edition*, Morgan Kaufman, 2009. Also available on line as *Appendix A: Assemblers, Linkers, and the SPIM Simulator* at http://pages.cs.wisc.edu/~larus/HP_AppA.pdf.

(Lesk, 1975) M. E. Lesk, “Lex - a lexical analyzer generator,” *Computing Science Technical Report 39*, AT&T Bell Laboratories, Murray Hill, NJ, 1975.

(Lesk and Schmidt) M. E. Lesk and E Schmidt, “Lex -- a lexical analyzer generator,” at <http://dinosaur.compilertools.net/lex/index.html>. An open-source implementation, originally written by Vern Paxton and called Flex, may be found at <http://flex.sourceforge.net>.

(Lewis and Stearns, 1968) P Lewis and R Stearns. “Syntax-directed Transduction.” *Journal of the ACM*, 15(3), 1968.

(Lewis, Rosenkrantz and Stearns, 1976) P Lewis, D Rosenkrantz and R Stearns. *Compiler Design Theory*. Addison-Wesley, 1976.

(Lindholm and Yellin, 1999) Tim Lindholm and Frank Yellin. *The Java TM Virtual Machine Specification (Second Edition)*. Prentice Hall (PTR), April 14, 1999. Also on the WWW at <http://java.sun.com/docs/books/vmspec/index.html>.

(Link, 2003) Johannes Link. *Unit Testing in Java: How Tests Drive the Code*. Morgan Kaufmann, April 2003.

(Linz, 2006) Peter Linz. *An Introduction to Formal Languages and Automata (Fourth Edition)*. Jones & Bartlett, February 15, 2006.

(Lippert, 2010) Eric Lippert. “How Many Passes?” *MSDN Blogs, Fabulous Adventures in Coding* – Eric Lippert’s blog at <http://blogs.msdn.com/b/ericlippert/archive/2010/02/04/how-many-passes.aspx>. 2010.

(Microsoft, 2011) *Microsoft .NET home page* on the WWW at <http://www.microsoft.com/net>.

(Mössenböck, 2000) Hanspeter Mössenböck. *Adding static single assignment form and a graph coloring register allocator to the Java HotSpot Client Compiler*. Technical Report 15, Institute for Practical Computer Science, Johannes Kepler University Linz, Linz, 2000.

Bibliography - 5

(MS C#, 2011) *Microsoft Visual C# Developer Center* on the WWW at <http://msdn.microsoft.com/en-us/vcsharp/aa336706>.

(MSDN Library, 2010) MSDN Library. *Expression Trees (C# and Visual Basic)* on the WWW at <http://msdn.microsoft.com/en-us/library/bb397951.aspx>.

(MSDN Library, 2011) *Overview of the .NET Framework*. On the WWW at <http://msdn.microsoft.com/en-us/library/a4t23ktk.aspx>.

(Muchnick, 1997) Steven S. Muchnick. *Advanced Compiler Design and Implementation*. Morgan Kaufmann, 1997.

(Naur, 1963) Peter Naur. *Revised Report on the Algorithmic Language ALGOL 60*. Communications of the ACM, Vol. 6, No. 1, Jan 1963.

(Notario, 2004) David Notario. *The CLR x86 JIT, An Overview*. MSDN Blogs, David Notario's WebLog, 26 Oct 2004, on the WWW at <http://blogs.msdn.com/b/davidnotario/archive/2004/10/26/247792.aspx>.

(Oracle, 2004) *Prototype for JSR014: Adding Generics to the Java[tm] Programming Language v. 2.4*. Oracle, 2004. On the WWW at http://java.sun.com/developer/earlyAccess/adding_generics/.

(Oracle Arch., 2010) *The Java HotSpot Performance Engine Architecture*. Technical White Paper, Oracle 2010. On the WWW at <http://www.oracle.com/technetwork/java/whitepaper-135217.html>.

(Oracle OSR, 2010) *The Java HotSpot Performance Engine: On-Stack Replacement Example*. Oracle, 2010. On the WWW at <http://java.sun.com/developer/technicalArticles/Networking/HotSpot/onstack.html>.

(Oracle, 2011) *Oracle Java SE HotSpot at a Glance*. Oracle, 2011. On the WWW at <http://www.oracle.com/technetwork/java/javase/tech/index-jsp-136373.html>.

(Paleczny, Click, Vick, 2001) Paleczny, C. Click, C. Vick. "The Java HotSpot™ Server Compiler." *The 2001 USENIX Java™ Virtual Machine Symposium*, Monterey, California, USA, April 23–24, 2001.

(Poletto et al, 1997) Massimiliano Poletto, Dawson R. Engler, M. Frans Kaashoek. "tcc: a system for fast, flexible, and high-level dynamic code generation." *Proceedings of the ACM SIGPLAN 1997 conference on Programming language design and implementation*, pp 109-121. ACM Press, 1997.

Bibliography - 6

(Poletto and Sarkar, 1999) Massimiliano Poletto, Vivek Sarkar. “Linear scan register allocation.” *ACM Transactions on Programming Languages and Systems (TOPLAS)*, Volume 21, Issue 5, pp 895-913. ACM Press, 1999.

(Rainsberger, 2004) J.B. Rainsberger. *JUnit Testing: Practical Methods for Programmer Testing*. Manning Publications, July 15, 2004.

(Richter, 2002) Jeffrey Richter. *Applied Microsoft .NET Framework Programming*. Microsoft Press, 2002.

(Richter, 2005) Jeffrey Richter. *.NET Questions regarding JIT compiler/strong-naming security*. 2005. Jeffrey Richter’s blog at <http://www.wintellect.com/cs/blogs/jeffreyr/archive/2005/07/25/net-questions-regarding-jit-compiler-strong-naming-security.aspx>.

(Richter, 2010) Jeffrey Richter. *CLR via C# (Third Edition)*. Microsoft Press, 2010.

(Sethi, 1973) Ravi Sethi. “Complete register allocation problems.” *STOC ’73 Proceedings of the fifth annual ACM symposium on Theory of Computing*, ACM, New York, NY, 1973.

(Shudo, 2005) Kazuyuki Shudo. *Performance Comparison of Java/.NET Runtimes*. Available via www.shudo.net/jit/perf.

(Sipser, 2005) Michael Sipser. *Introduction to the Theory of Computation (Second Edition)*. Course Technology, February 15, 2005.

(Slonneger, 1994) Kenneth Slonneger. *Formal Syntax and Semantics of Programming Languages: A Laboratory-Based Approach*. Addison Wesley Longman, September 1994. This text is out of print but you may obtain a copy by contacting the author at slonnegr@cs.uiowa.edu.

(van der Spek, Plat and Pronk, 2005) P van der Spek, N Plat and C Pronk. “Syntax error repair for a Java-based parser generator.” *ACM SIGPLAN Notices* 40(4), April 2005, pp 47-50.

(Strachey, 1967) Christopher Strachey. “Fundamental concepts in programming languages.” *Notes for the International Summer School in Computer Programming*, Copenhagen, 1967.

(Traub, 1998) Omri Traub, Glenn Holloway and Michael D. Smith. “Quality and speed in linear-scan register allocation.” *ACM SIGPLAN 1998 Conference on Programming Language Design and Implementation*, pp. 142-151, June 1998.

Bibliography - 7

(Turner, 1977) Turner D.A. “Error diagnosis and recovery in one pass compilers.” *Information Processing Letters*, Vol 6, No 4, 1977.

(Waite and Goos, 1984) William Waite and Gerhard Goos. *Compiler Construction*. Springer Verlag, 1984.

(Whittaker, 2002) James Whittaker. *How to Break Software: a Practical Guide to Testing*. Addison Wesley, 2002.

(Wielaard, 2005) Mark Wielaard. *GCC - past, present, and future*. LWN.net, April 2005. Available via <http://lwn.net/Articles/130796/>.

(Wilson, 1994) Paul R. Wilson. “Uniprocessor Garbage Collection Techniques.” *Technical report*, University of Texas, January 1994. Expanded version of a *IWMM92* paper. Postscript available at <ftp://ftp.cs.utexas.edu/pub/garbage/bigsurv.ps> .

(Wilson and Kesselman, 2000) Steve Wilson and Jeff Kesselman. *Java Platform Performance: Strategies and Tactics*. Prentice Hall, 2000.

(Wimmer, 2004) Christian Wimmer. *Linear Scan Register Allocation for the Java HotSpot Client Compiler*. Master of Science Thesis, Institute for System Software, Johannes Kepler University Linz, Linz, August 2004.

(Wimmer and Franz, 2010) Christian Wimmer and Michael Franz. “Linear Scan Register Allocation on SSA Form.” *Proceedings of 2010 the International Symposium on Code Generation and Optimization*, April 24-28, 2010, Toronto, Ontario, pp. 170–179, 2010.