

1. Record Nr.	UNINA9910999689703321
Autore	Mössenböck Hanspeter
Titolo	Compiler Construction : Fundamentals and Applications / / by Hanspeter Mössenböck
Pubbl/distr/stampa	Cham : , : Springer Nature Switzerland : , : Imprint : Springer, , 2025
ISBN	3-031-84813-6
Edizione	[1st ed. 2025.]
Descrizione fisica	1 online resource (XIII, 251 p. 128 illus., 3 illus. in color.)
Disciplina	005.45
Soggetti	Compilers (Computer programs) Software engineering Programming languages (Electronic computers) Compilers and Interpreters Software Engineering Programming Language
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Nota di contenuto	1. Overview -- 2. Lexical Analysis -- 3. Syntax Analysis -- 4. Attributed Grammars -- 5. Symbol Table -- 6. Code Generation -- 7. The Compiler Generator Coco/R -- 8. Bottom-up Syntax Analysis.
Sommario/riassunto	This textbook covers the fundamentals of compiler construction, from lexical analysis and syntax analysis to semantic processing and code generation. As a running example, a compiler for a simple Java-like programming language (MicroJava) is described and developed. It generates executable bytecode similar to Java bytecode. Other topics include the description of translation processes using attributed grammars and the use of a compiler generator to automatically generate the core parts of a compiler. For syntax analysis, the book concentrates on top-down parsing using recursive descent, but also describes bottom-up parsing. All code examples are presented in Java. A companion web page contains a full set of PowerPoint slides for an introductory compiler course, sample solutions for more than 70 exercises provided at the end of each chapter to practice and reinforce the content of that chapter, and the full source code of the MicroJava compiler as well as other code samples. In addition, the open-source

compiler generator Coco/R described in the book is provided as an executable and in source code. The book targets both students of Computer Science or related fields as well as practitioners who want to apply basic compiling techniques in their daily work, e.g., when crafting software tools. It can be used as a textbook for an introductory compiler course on which more advanced courses on compiler optimizations can be based.
