



Compiler Design

by **B. Ankayarkanni, D. Usha Nandini, A. Mary Posonia, V. Ulagamuthalvi**

ISBN-13	9789355385000
ISBN-10	9355385005
Edition	First
Format	Both
Publishing Year	2026
Categories	New Arrivals, Computer Science Engineering, Book Store, Sathyabama Series
Original Price	₹248.00
Discounted Price	₹210.80
Stock Status	In Stock (5 available)

Description

Compiler Design offers a comprehensive and organized introduction to the foundational principles, techniques, and practical tools necessary for constructing a compiler. Serving as an essential bridge between human-readable high-level programming languages and machine instructions, this book aims to unravel the complexity of the translation process and demystify compiler construction for both students and academics.

The book systematically guides readers through the entire compiler lifecycle, detailing each major phase: Lexical Analysis, Syntax Parsing, Semantic Analysis, Intermediate Code Generation, Code Optimization, and Final Code Generation. It adopts an approachable, academic tone, striving to simplify both fundamental ideas and advanced concepts by integrating theoretical foundations with practical implementations. For enhanced comprehension, the material is reinforced with a combination of theoretical discussions and numerous solved numerical problems. Furthermore, a strong emphasis is placed on practical skill development, featuring dedicated coverage of industry-standard tools like LEX and YACC, ensuring the target audience—computer science students and professors—gains hands-on experience in building real-world compilers. This indispensable resource provides a clear pathway to mastering the art and science of transforming source code into efficient, executable programs.

Salient Features:

- **Systematic Phase Coverage:** Presents the compiler's structure through distinct, organized units, detailing the functionality of the analysis-synthesis model from scanning to code generation.
- **Practical Tool Integration:** Provides detailed sections on using LEX and YACC (Yet Another Compiler Compiler), giving students hands-on experience in generating practical Lexical Analyzers and Parsers.
- **Comprehensive Parsing Methods:** Explores a full range of parsing techniques, including Top-Down (LL(1)) and sophisticated Bottom-Up (LR, SLR) parsing, complete with error handling and recovery strategies.
- **Intermediate Code Representation:** Dedicated unit focusing on Syntax-Directed Translation (SDT), implementing three-address code, and handling complex programming constructs like procedure calls and switch-case statements.
- **Advanced Optimization Techniques:** Covers crucial optimization strategies like Data Flow Analysis, Loop Optimization, and the use of Directed Acyclic Graphs (DAGs) for efficient, high-speed target code generation.
- **Efficient Code Generation:** Discusses critical implementation issues such as managing Run-Time Storage, optimal Register Allocation and Assignment, and the final conversion of assembly code to machine code.
- **Formal Language Foundation:** Clearly explains the role of Formal Languages, covering Regular Expressions and both Deterministic and Non-Deterministic Finite Automata (DFA/NFA) conversions in lexical design.

About the Author(s)

B. Ankayarkanni

D. Usha Nandini

A. Mary Posonia

V. Ulagamuthalvi