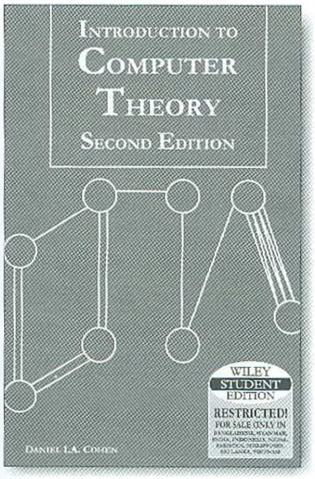

INTRODUCTION TO COMPUTER THEORY

By Daniel I.A. Cohen

John Wiley & Sons (Asia) Pvt. Ltd.
Second Edition, 2006

Pages : 634, Price Rs. 384/-

Reviewed By:
Mr. Lalit Sharma
Faculty

Delhi Institute of Advanced Studies

Leaving aside the obvious worth of knowledge, the terminology, notations and techniques of computer theory are necessary in teaching Computer Design, Artificial Intelligence, the analysis of algorithms and so forth. Of all programming skills that students learn, two of the most important things are the abilities to recognize and manipulate context-free grammars and to understand the power of recursive interaction of parts of procedures. In order to have sound foundation, each advance course has to begin at the level of defining rules of productions and derivations. Any career that might be pursued by a Computer Science graduate will make significant use of subject matter of this book.

Usually computer science specializations do not speak the language of mathematical symbolism fluently, nor it is important at that level. The value of mathematical iconography is that it enables professionals to perform their research and communicate their results more efficiently. This book is not only meant for graduate and post graduate students but also for research scholars.

Basically this book is written for students without prerequisite of any kind of background. Every mathematical concept used is introduced from scratch. Extensive examples & illustrations speak out everything in detail to avoid any possibility of confusion. The students are encouraged to read at their own pace and depth required.

The book starts with the basic utilities like definition of language. Then it proceeds to explain how the alphabets can be converted into strings, strings into language sets, properties of sets & languages. The graphical representation is highlighted in the form of Finite Automata (deterministic & non-deterministic) and Transition graphs. In the second chapter, the concept of regular expression, regular languages are demonstrated through Kleen's Theorem, regular sets, Moore & Mealy Machine & reduction of states in FA.

The chapters of the book contain a vast study of grammars, including the type of grammar, their properties, closure properties, plumping lemma, string matching, representation of CFG's & CSG's through pushdown automata and LBA. As grammar is most important in understanding the context of any

language, author tried hard to locate the actual process going on in programming background which is very useful in terms of compiler design. The concluding chapters of the book contains a study of advanced topic of automata theory like Turing Machine, Computable Functions, Post System, their Decidability, Universal Turing Machine, Post Correspondence Problem, Halting Problem of Turing Machine etc. which provide the basis for computable device and algorithm verification tools. These complex concepts are explained in very easy language.

We shall study different types of theoretical machines that are mathematical models for actual physical processes throughout this book. By considering the possible inputs on which these machines can work, we can analyze their various strengths & weaknesses. The book also highlight the most powerful possible machine. Along with that, we shall begin to understand the concept of computability, which is the foundation of further research in this field. This book extends further to such topics as complexity & verification in great depth. Complexity also include complexity classes, namely: P & NP completeness that are also explored. So, it enables to compute the complexity of given algorithm.

Ideally, a textbook should begin at the level of understanding of the students taking the course. However, there are some fanatics who have the grandiose notion that to be a great teacher is to stuff more material into a course than their students can learn. This book provides the reflective teaching approach for such teachers.

Overall, this book is of tremendous utility to those who want to understand the internal functionality of computability. It reflects an appreciable degree of lucidity & readability. This book is unabashedly easy to read. It is intentionally slow-paced and repetitive. The amount of efforts that must have gone into this very professional exercise is appreciable.

In this book no pretense is made to encyclopedic completeness. NP completeness, primitive and partial recursion, artificial intelligence nor renaissance architecture have been introduced in this book.

This book may not be the best but it is designed with good scientific intensions & sincere concern for those interested in learning. It is sure to deliver a much greater value to reader compared to price he/ she pays for it.