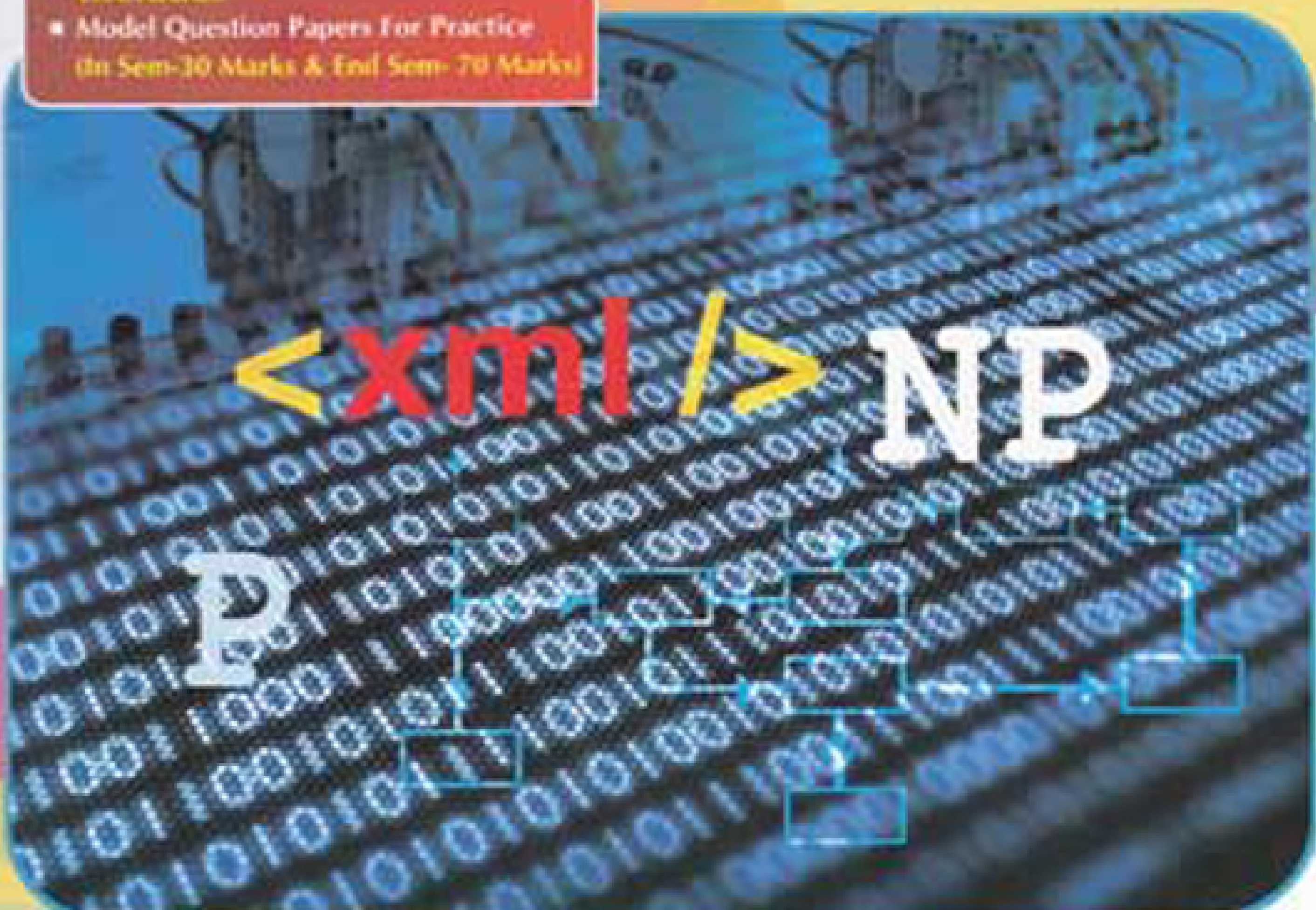
According to New Revised Credit System Syllabus

SPPU

Third Year Degree Course In
COMPUTER ENGINEERING (Sem - I)

THEORY OF COMPUTATION

Includes
- Model Question Papers For Practice
(In Sem-30 Marks & End Sem-70 Marks)



Mrs. SWATI K. BHAVSAR
Ms. ABOLEE H. PATIL
Mrs. AKSHITA S. CHANCHLANI
Mrs. NIKHITA A. PATIL(NERKAR)

www.pragationline.com
www.facebook.com/niralibooks

NIRALI PRAKASHAN
ADVANCEMENT OF KNOWLEDGE

A TEXT BOOK OF

THEORY OF COMPUTATION

FOR

SEMESTER – I

THIRD YEAR DEGREE COURSE IN COMPUTER ENGINEERING

Strictly According to New Revised Credit System Syllabus of Savitribai Phule Pune University

(w.e.f. June 2017)

Mrs. SWATI K BHAVSAR
ME (Comp Engg)
Assistant Professor,
Computer Engg. Deptt.,
Matoshri College of Engg. & Research Centre
Nashik.

Mrs. AKSHITA S CHANCHLANI
B.Tech. (VJTI) ME (IT)
Assistant Professor,
Computer Engg. Deptt.,
Sinhgad Acadamy of Engineering,
Kondhwa, Pune.

Ms. ABOLEE H. PATIL
ME (Comp.)
Assistant Professor,
Computer Engg. Deptt.,
MIT College of Engineering,
Kothrud, Pune.

Mrs. NIKHITA A. PATIL (NERKAR)
ME (Comp.)
Assistant Professor,
Computer Engg. Deptt.,
RMD Sinhgad School of Engineering
Warje, Pune.

Price ₹ 400.00

NIRALI PRAKASHAN
ADVANCEMENT OF KNOWLEDGE

N4186

THEORY OF COMPUTATION (TE COMP.) **ISBN : 978-93-86700-08-7**

First Edition : June 2017

Published By :
NIRALI PRAKASHAN
Abhyudaya Pragati, 1312, Shivaji Nagar,
Off J.M. Road, Pune – 411005
Tel - (020) 25512336/37/39, Fax - (020) 25511379
Email : niralipune@pragationline.com

Polyplate

Printed By :
YOGIRAJ PRINTERS AND BINDERS
Survey No. 10/1A, Ghule Industrial Estate
Nanded Gaon Road
Nanded, Pune - 411041
Mobile No. 9404233041/9850046517

Dedicated to…

“Our Parents”

… Authors

PREFACE

It gives us great pleasure to present the book **'Theory of Computation'** for the students of Third Year Degree Course in Computer Engineering of the Savitribai Phule Pune University. This book is strictly as per the new revised syllabus 2015 Pattern with effect from the Academic Year (2017-18).

As per New Revised Examination Scheme which has been implemented from this academic year, In-semester assessment carries 30 marks over first three units and End Semester Examination carries 70 marks over entire syllabus out of which first three units will carry 20 marks and units 4, 5, 6 will carry 50 marks. The theory course will have 4 credits.

The book is written such that all the basic concepts are explained in simplified manner. It is presented in a more conceptual manner rather than mathematical, as required by the new examination system. It is our objective to keep the presentation systematic, consistent, intensive and clear through explanatory notes and figures. Main feature of this book is, **Complete Coverage** of the New Credit System Syllabus with large number of Worked Solved Examples, Exercises, **Model Question Papers of In Sem.** and **End Sem. Exams.**

We are sure that this book will cater to all needs of students for this subject.

We would like to our sincere thanks to **Management** and **Dr. G.K. Kharate** (Principal, MCOERC Nashik), **Dr. K.P. Patil** (Principal SAE Kondhwa), **Dr. C.B. Bangal** (Principal RMDSSOE Pune.), **Dr. M.S. Nagmode** (Principal, MITCOE Pune), **Prof. B.B. Gite** (HOD SAE Kondhwa), **Mrs. V.M. Lomte** (HOD Comp. Engg. Dept. RMDSSOE, Pune.) **Mrs. R.K. Bedi** (HOD Comp. Engg. Dept. MITCOE Pune), for their untiring support in our work.

Our Special thanks to **Dr. (Mrs.) Varsha H. Patil** Vice Principal, Professor and Head, Computer Engg. Deptt. MCOERC, Nashik) for her valuable guidance, timely advice and continuous encouragement.

We also take this opportunity to express our sincere thanks to Shri. Dineshbhai Furia, Shri. Jignesh Furia, Mrs. Nirali Verma, Shri. M. P. Munde and entire team of Nirali Prakashan namely Mrs. Deepali Lachake (Co-ordinator), who really have taken keen interest and untiring efforts in publishing this text.

The advice and suggestions of our esteemed readers to improve the text are most welcomed, and will be highly appreciated.

Pune **Authors**

SYLLABUS

Unit I : Formal Language Theory and Finite Automata (08 Hrs)

Introduction to Formal language, introduction to language translation logic, Essentials of translation, Alphabets and languages, Finite representation of language, Finite Automata (FA): An Informal Picture of FA, Finite State Machine (FSM), Language accepted by FA, Definition of Regular Language, Deterministic and Nondeterministic FA(DFA and NFA), epsilon- NFA, FA with output: Moore and Mealy machines -Definition, models, inter-conversion.

Case Study: FSM for vending machine, spell checker

Unit II : Regular Expressions (RE) (07 Hrs)

Introduction, Operators of RE, Building RE, Precedence of operators, Algebraic laws for RE, Conversions: NFA to DFA, RE to DFA Conversions: RE to DFA, DFA to RE Conversions: State/loop elimination, Arden's theorem Properties of Regular Languages: Pumping Lemma for Regular languages, Closure and Decision properties.

Case Study: RE in text search and replace

Unit III : Context Free Grammars (CFG) and Languages (07 Hrs)

Introduction, Regular Grammar, Context Free Grammar- Definition, Derivation, Language of grammar, sentential form, parse tree, inference, derivation, parse trees, ambiguity in grammar and Language- ambiguous Grammar, Simplification of CFG: Eliminating unit productions, useless production, useless symbols, and Є-productions, Normal Forms- Chomsky normal form, Greibach normal form, Closure properties of CFL, Decision properties of CFL, Chomsky Hierarchy, Application of CFG: Parser, Markup languages, XML and Document Type Definitions.

Case Study- CFG for Palindromes, Parenthesis Match,

Unit IV : Turing Machines (08 Hrs)

Turing Machine Model, Representation of Turing Machines, Language Acceptability by Turing Machines, Design of TM, Description of TM, Techniques for TM Construction, Variants of Turing Machines, The Model of Linear Bounded Automata , TM & Type 0 grammars, TM's Halting Problem.

Unit V : Pushdown Automata (07 Hrs)

Basic Definitions, Equivalence of Acceptance by Finite State & Empty stack, PDA & Context Free Language, Equivalence of PDA and CFG, Parsing & PDA: Top-Down Parsing, Top-down Parsing Using Deterministic PDA, Bottom-up Parsing, Closure properties and Deterministic PDA.

Unit VI : Undecidability and Intractable Problems (07 Hrs)

A Language that is not recursively enumerable, An un-decidable problem that is RE, Post Correspondence Problem, The Classes P and NP : Problems Solvable in Polynomial Time, An Example: Kruskal's Algorithm, Nondeterministic Polynomial Time, An NP Example: The Traveling Salesman Problem, Polynomial-Time Reductions NP Complete Problems, An NP-Complete Problem: The Satisfiability Problem, Tractable and Intractable, Representing Satisfiability, Instances, NP Completeness of the SAT Problem, A Restricted Satisfiability Problem: Normal Forms for Boolean Expressions, Converting Expressions to CNF, The Problem of Independent Sets, The Node-Cover Problem.

CONTENTS

Unit I

FORMAL LANGUAGE THEORY AND FINITE AUTOMATA

1.1 INTRODUCTION TO FORMAL LANGUAGE

- A language is a subset of all possible strings formed from a given set of symbols.
- In general, a natural language is one that has evolved over time for the purpose of human communication.
- Examples include English, German, and Hindi. Such languages continue to evolve without regard for formal grammatical rules.
- Natural languages rarely conform to simple or obvious a grammatical rules.
- In contrast to natural languages, formal languages are defined by, and hence rigorously conform to, pre established rules.
- Examples mathematical languages such as algebra and prepositional logic. It is because of this adherence to rules that the construction of efficient computerized translators for programming languages is possible, whereas it is the lack of adherence to established rules that makes it difficult to construct a translator for a natural language.

Alphabet : A non-empty, finite set of symbols from which the strings to be analyzed are constructed, denoted as Σ.

Language : A subset of Σ* (collection of all strings of finite length including the empty string that can be constructed alphabet Σ being used) is called as language over Σ.

1.2 INTRODUCTION TO LANGUAGE TRANSLATION LOGIC

- A translator is a computer program that performs translation of a program written in a given programming language into a functionally equivalent program in a different ways computer language.
- But while translating it takes care that it should not loose the functional or logical structure of original code.
- It mainly consist of translation between high level languages e.g. C, C++, Java ink low level languages such as, machine code, assembly language and byte code.
- As the computer can only understand programs defined using machine code.
- There are three types of system s/w are used for translating the code which is written by programmer into a form which is understandable to machine are :

1. **Assembler :** It converts an assembly program into machine language.

Source code ⟶ Assembler ⟶ Executable code

2. **Compiler :** It translates a high level program into machine code.

Source code ⟶ Compiler ⟶ Executable code

3. Interpreter: It translates high level source code into executable code.

The difference between compiler and interpreter is that, it translates one line at a time and then executes it.

1.3 ESSENTIALS OF TRANSLATION

- A translator is very essential for a program which is written in a given programming language and to be converted into a functionally equivalent program.
- It should work in efficient way without losing functional or logical structure of original code.
- As computer does not understand the high level language, there is a necessity of translator to translate into machine level language.
- Semantic plays an important role for language translation. It is associated with meaning of language.

1.4 ALPHABETS AND LANGUAGES

1.4.1 An Alphabet

- An alphabet is a finite non-empty set of symbols. It is denoted by Σ. Symbols may be numbers or letters.
- For example,
  1) $\Sigma = \{0, 1\}$
  2) $\Sigma = \{a, b, c, \ldots\ldots z\}$
  3) $\Sigma = \{0, 1, 2, \ldots\ldots 9\}$

Power of an alphabet :

- If Σ is an alphabet, the set of all strings of a certain length 'k' from that alphabet is nothing but power of alphabet.

For example,

$\Sigma^0 = \{\in\}$

$\Sigma = \{0, 1\}$

$\Sigma^1 = \{0, 1\}$

$\Sigma^2 = \{00, 11, 01, 10\}$