

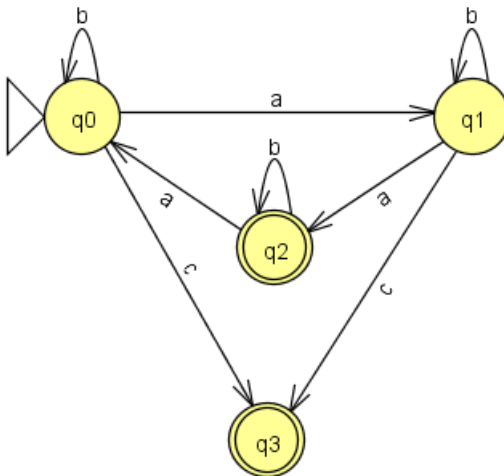


Digitally signed by
Technical Scientific Library,
TUM
Reason: I attest to the
accuracy and integrity of
this document

TECHNICAL UNIVERSITY OF MOLDOVA

Formal Languages and Finite Automata

Guide for practical lessons



**Chişinău
2022**

TECHNICAL UNIVERSITY OF MOLDOVA

Faculty of Computers, Informatics and Microelectronics

**SOFTWARE ENGINEERING AND AUTOMATICS
DEPARTMENT**

Formal Languages and Finite Automata

Guide for practical lessons

**Chişinău
Editura „Tehnica-UTM”
2022**

CZU 004.43+519.713(076)

C 61

Lucrarea a fost discutată și aprobată pentru editare la ședința Consiliului Facultății Calculatoare, Informatică și Microelectronică din 27.10.22, proces-verbal nr. 2.

This guide is intended for the students from the study program *Software Engineering*, Faculty of Computers, Informatics and Microelectronics, Technical University of Moldova, for the practical lessons at the course *Formal Languages and Automata*.

The guide includes six chapters, that address topics related to formal languages, finite automata, context free languages, pushdown automata, syntactic analysis and Turing machine. Theoretical approaches are presented together with practical examples of solving different exercises and each topic is provided with the list of practical tasks.

Authors: COJUHARI Irina, DUCA Ludmila, FIODOROV Ion

Redactor responsabil:

Recenzent: COJUHARI Elena, conf. univ., dr.

DESCRIEREA CIP A CAMEREI NAȚIONALE A CĂRȚII DIN RM

Cojuhari, Irina.

Formal Languages and Finite Automata: Guide for practical lessons/
Cojuhari Irina, Duca Ludmila, Fiodorov Ion; Technical University of
Moldova, Faculty of Computers, Informatics and Microelectronics,
Software Engineering and Automatics Department.

– Chișinău: Tehnica-UTM, 2022. – 120 p.: fig., tab.

Aut. indicați pe vs. f. de tit. – Referințe bibliogr.: p. 120 (10 tit.). – 50 ex.

Bun de tipar 07.11.22

Formatul hârtiei 60x84 1/16

Hârtie ofset. Tipar RISO.

Comanda nr. 100

MD-2004, Chișinău, bd. Ștefan cel Mare și Sfânt, 168. UTM

MD-2045, Chișinău, str. Studenților, 9/9. Editura "Tehnica-UTM"

ISBN 978-9975-45-852-8

© UTM, 2022

Contents

1	WORDS AND LANGUAGES	5
1.1	Words and Alphabets	5
1.2	Languages	8
1.3	Grammar	10
1.4	Derivation Tree	12
1.5	Ambiguous Grammar	14
1.6	Chomsky Classification of Grammars	16
2.	FINITE AUTOMATA	21
2.1	Finite Automata	21
2.1.1	Deterministic Finite Automata.....	21
2.1.2	Nondeterministic Finite Automata with λ transitions.....	25
2.1.3	Conversion the NFA to DFA	30
2.1.4	Conversion λ - NFA to DFA	35
2.1.5	Equivalence of the Finite Automaton with Regular Grammar	38
2.1.6	Equivalence of the Regular Grammar with Finite Automata.....	41
2.1.7	Minimization of the DFA	43
2.2	Regular Expressions	47
2.3	Pumping Lemma.....	50
3.	CONTEXT FREE GRAMMAR	54
3.1	Elimination of λ – productions	54
3.2	Elimination of the Unit Productions	58

3.3 Elimination of the Inaccessible Symbols.....	60
3.4 Elimination of the Non-Productive Symbols.....	62
3.5 Chomsky Normal Form	64
3.6 Left Recursion.....	69
3.7 Greibach Normal Form.....	76
4. PUSHDOWN AUTOMATA	83
4.1 Pushdown Automata.....	83
4.2 Conversion of a Context Free Grammar into a PDA.....	87
5. SYNTACTIC ANALYSIS.....	91
5.1 Introduction.....	91
5.2 Bottom-up Parser	91
5.3 Top-Down Parsing.....	105
6. TURING MACHINE.....	114
REFERENCES.....	120

$$L = \{ 0^{2^n} \mid n \geq 0 \}.$$

4. Design the Turing Machine, which takes two input words generated with the alphabet $\{a, b, c\}$, separated by the symbol $\{\#\}$, and verifies whether they are the same. For example, given the input $abc\#abc$, in this case the Turing Machine would inform that both words are the same.
5. Design the Turing Machine, which generates a copy of a string with symbols $\{X, Y, Z\}$. For example, given the input “ XYZ ”, the resulting input tape would be “ $XYZXYZ$ ”.
6. Design the Turing Machine, which compares two words a and b generated by the alphabet $\{0, 1\}$ and Turing Machine should recognize if words: $a=b$, $a<b$ or $a>b$.
7. Design the Turing machine that looks for the three consecutive occurrences of letter a in the input string generated by the alphabet $\{a, b, c\}$. If it is found, the input is accepted, otherwise it is rejected.
8. Design the Turing Machine, which takes input word generated by the alphabet $\{a, b\}$ and as output insert a blank between each of the input symbols.