

[Crack GATE 2026](#)[Explore GfG Courses](#)**[Automata Tutorial](#)**[Automata \\_  
Introduction](#)[Regular Expression  
and Finite  
Automata](#)[CFG](#)[PDA \(Pushdown  
Automata\)](#)[Turing Machine](#)[Decidability](#)[TOC Interview  
preparation](#)

# Automata Tutorial

Last Updated : 30 Jan, 2025

Automata Theory is a branch of the Theory of Computation. It deals with the study of abstract machines and their capacities for computation. An abstract machine is called the automata. It includes the design and analysis of automata, which are mathematical models that can perform computations on strings of symbols according to a set of rules.

## Why we study Theory of Computation?

- **Regular Expressions (RE)** : Used for pattern matching in Linux/Unix command prompt, programming languages and XML/DTD to describe structure.
- **Finite Automata in Modeling Systems** : Used in designing and checking models and electronic circuits that operate based on certain rules.

[Skip to content](#)



- **Context-Free Grammars (CFG)** : Used in Compiler / Programming Language design to describe syntax and natural language processing to describe structure.
- **Mathematical Models** : Mathematical understanding of computing devices by mathematically modeling them.
- **Building Block for Quantum Computing**: Turing Machines (we study in this subject) are considered a fundamental building block for understanding quantum computation models.
- **Optimizing Algorithm Efficiency** : Helps classify problems based (e.g., **P**, **NP**, **NP-complete**, and **NP-hard**), proving that some problems have no efficient solutions.
- **Understanding Computability** : Study of which problems can be solved using algorithms, essentially defining the boundaries of what a computer can calculate.. Problems like the "Halting Problem" which are demonstrably impossible to solve with a general algorithm.

Please refer [Why we Study Theory of Computation?](#) for details.

[Skip to content](#)

## Automata – Introduction

1. [Introduction](#)
2. [Chomsky Hierarchy](#)
3. [Applications of various Automata](#)

## Regular Expression and Finite Automata

1. [Finite Automata Introduction](#)
2. [Arden's Theorem](#)
3. [L-graphs and what they represent](#)
4. [Hypothesis \(language regularity\) and algorithm \(L-graph to NFA\)](#)
5. [Regular Expressions, Regular Grammar and Regular Languages](#)
6. [How to identify if a language is regular or not](#)
7. [Designing Finite Automata from Regular Expressions](#)
8. [Star Height of Regular Expression and Regular Language](#)
9. [Generating regular expression from finite automata](#)
10. [Designing Deterministic Finite Automata \(Set 2\)](#)
11. [NFA to DFA Conversion](#)

[Skip to content](#)

12. [Program to Implement NFA with epsilon move to DFA Conversion](#)
13. [Minimization of DFA](#)
14. [Kleene's Theorem Part-1](#)
15. [MEALY and MOORE Machines](#)
16. [Difference between Mealy machine and Moore machine](#)
17. [Problems on Finite Automata](#)
18. [Operations on DFA](#)

[>> Quiz on Regular Languages and Finite Automata](#)

## CFG (Context Free Grammar)

1. [Relationship between grammar and language](#)
2. [Simplifying Context Free Grammars](#)
3. [Closure Properties of Context Free Languages\(CFL\)](#)
4. [Union & Intersection of Regular languages with CFL](#)
5. [Converting Context Free Grammar to Chomsky Normal Form](#)
6. [Converting Context Free Grammar to Greibach Normal Form](#)
7. [Pumping Lemma](#)
8. [Check if the language is Context Free or Not](#)

[Skip to content](#)

9. [Ambiguity in Context Free Grammar](#)
10. [Operator grammar and precedence parser](#)
11. [Context-sensitive Grammar \(CSG\) and Language \(CSL\).](#)

## PDA (Pushdown Automata)

1. [Pushdown Automata](#)
2. [Pushdown Automata Acceptance by Final State](#)
3. [Detailed Study of PushDown Automata](#)
4. [Problems on Pushdown Automata](#)

[>> Quiz on Context Free Languages and Pushdown Automata](#)

## Turing Machine

1. [Turing Machine](#)
2. [Halting Problem](#)
3. [Theory of Computation | Applications of various Automata](#)
4. [Turing Machine as Comparator](#)
5. [Problems on Turing Machine](#)

[>> Quiz on Turing Machines and Recursively Enumerable Sets](#)

[Skip to content](#)

## Decidability

1. [Decidable and undecidable problems](#)
2. [Decidability](#)
3. [Undecidability and Reducibility](#)
4. [NP-Completeness | Set 1 \(Introduction\)](#)
5. [Proof that Hamiltonian Path is NP-Complete](#)
6. [Proof that vertex cover is NP complete](#)
7. [Computable and non-computable problems](#)

[>> Quiz on Undecidability](#)

## Quick Links

- [Last Minute Notes\(LMNs\)](#)
- [‘Quizzes’ on Theory Of Computation !](#)
- [Recent Articles on Theory Of Computation](#)

[Comment](#)[More info](#)[Advertise with us](#)

### Next Article

Introduction to Theory  
of Computation

## Similar Reads

[Why Do We Study Theory of](#) [Skip to content](#)

Theory of Computation (ToC) is the study of how computers solve problems using mathematical model...

15+ min read

## Introduction to Theory of Computation

Automata theory, also known as the Theory of Computation, is a field within computer science and...

15+ min read

## Applications of various Automata

Automata is a machine that can accept the Strings of a Language  $L$  over an input alphabet  $\Sigma$ . So far we are...

15+ min read

## Practice problems on finite automata

Que-1: Draw a deterministic and non-deterministic finite automate which accept 00 and 11 at the end of ...

10 min read

## Regular Expressions, Regular Grammar and Regul...

To work with formal languages and string patterns, it is essential to understand regular expressions, regular...

[Skip to content](#)

15+ min read

### Designing Deterministic Finite Automata (Set 5)

Prerequisite: Designing finite automata In this article, we will see some designing of Deterministic Finite...

13 min read

### Designing Finite Automata from Regular Expressio...

In this article, we will see some popular regular expressions and how we can convert them to finite...

15+ min read

### Introduction To Grammar in Theory of Computation

In Theory of Computation, grammar refers to a formal system that defines how strings in a language are...

15+ min read

### Computer Organization and Architecture Tutorial

In this Computer Organization and Architecture Tutorial, you'll learn all the basic to advanced concepts like...

15+ min read

[Skip to content](#)



# Arden's Theorem in Theory of Computation

Arden's Theorem is a fundamental result in the Theory of Computation used to solve regular expressions fro...

15+ min read



**Corporate & Communications Address:**  
A-143, 7th Floor, Sovereign Corporate Tower, Sector- 136, Noida, Uttar Pradesh (201305)

**Registered Address:**  
K 061, Tower K, Gulshan Vivante Apartment, Sector 137, Noida, Gautam Buddh Nagar, Uttar Pradesh, 201305



Advertise with us

## Company

- About Us
- Legal
- Privacy Policy
- Careers
- In Media
- Contact Us
- GfG Corporate Solution
- Placement
- Training Program

## Explore

- Job-A-Thon
- Hiring Challenge
- GfG Weekly Contest
- Offline Classroom Program
- DSA in JAVA/C++
- Master System Design
- Master CP
- GeeksforGeeks Videos

## Languages

- Python
- Java
- C++
- PHP
- GoLang
- SQL
- R Language
- Android Tutorial

## DSA

- Data Structures
- Algorithms
- DSA for Beginners
- Basic DSA Problems
- DSA Roadmap
- DSA Interview Questions
- Competitive Programming

## Data Science & ML

- Data Science With Python
- Data Science For Beginner
- Machine Learning
- ML Maths
- Data Visualisation
- Pandas
- NumPy
- NLP
- Deep Learning

## Web Technologies

- HTML
- CSS
- JavaScript
- TypeScript
- ReactJS
- NextJS
- NodeJs
- Bootstrap
- Tailwind CSS

## Python Tutorial

- Python Programming Examples
- Django Tutorial

## Computer Science

- GATE CS Notes
- Operating Systems

## DevOps

- Git
- AWS
- Docker
- Kubernetes
- Azure

## System Design

- High Level Design
- Low Level Design
- UML Diagrams

## School Subjects

- Mathematics
- Physics
- Chemistry
- Biology

## Databases

- SQL
- MYSQL
- PostgreSQL
- PL/SQL
- MongoDB

Skip to content

Python Projects	Computer	GCP	Interview Guide	Social Science
Python Tkinter	Network	DevOps	Design Patterns	English Grammar
Web Scraping	Database	Roadmap	OOAD	
OpenCV Tutorial	Management		System Design	
Python Interview	System		Bootcamp	
Question	Software		Interview	
	Engineering		Questions	
	Digital Logic			
	Design			
	Engineering			
	Maths			

Preparation  
Corner

Company-Wise  
Recruitment  
Process  
Aptitude  
Preparation  
Puzzles  
Company-Wise  
Preparation

More  
Tutorials

Software  
Development  
Software Testing  
Product  
Management  
Project  
Management  
Linux  
Excel  
All Cheat Sheets

Machine  
Learning/  
Data Science

Complete  
Machine  
Learning & Data  
Science Program  
- [LIVE]  
Data Analytics  
Training using  
Excel, SQL,  
Python &  
PowerBI - [LIVE]  
Data Science  
Training  
Program - [LIVE]  
Data Science  
Course with IBM  
Certification

Programming  
Languages

C Programming  
with Data  
Structures  
C++  
Programming  
Course  
Java  
Programming  
Course  
Python Full  
Course

Clouds/  
Devops

DevOps  
Engineering  
AWS Solutions  
Architect  
Certification  
Salesforce  
Certified  
Administrator  
Course

GATE 2026

GATE CS Rank  
Booster  
GATE DA Rank  
Booster  
GATE CS & IT  
Course - 2026  
GATE DA Course  
2026  
GATE Rank  
Predictor

