UNIT V: Turing Machine – Complete Guide

1. Turing Machine: Definition & Model

A **Turing Machine (TM)** is a 7-tuple $(Q, \Sigma, \Gamma, \delta, q_0, B, F)$, where:

- Q: finite set of states
- Σ: input alphabet
- Γ: tape alphabet (includes a blank symbol B)
- δ: transition function $Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$
- q_0: initial (start) state
- B: blank symbol
- F: set of final (accepting) states

A TM is an abstract computing device consisting of an infinite tape, a tape head that can read/write/shift left/right, and a finite control unit that manages transitions. TM models all computation and proves foundational to the theory of computability[1][2].

Video Resource: Introduction to Turing Machine — Sudhakar Atchala
Watch here

GeeksforGeeks Reference:

- Turing Machine in TOC
- Turing Machine Introduction (Tutorialspoint)

---

2. Representation of TMs: Instantaneous Descriptions, Transition Tables, Diagrams

- **Instantaneous Description (ID):** Snapshot of TM at time $(q, w_1 a w_2)$, where q is state, a is current symbol, $w_1 w_2$ is tape.
- **Transition Table:** Specifies next state, tape symbol to write, and movement for each $(\text{state}, \text{symbol})$ pair[2][3].
- **Transition Diagrams:** Graphical visualization: nodes for states, labeled edges with input/output/move.

Video Resource: Turing Machine Formal Representation
Watch here

GeeksforGeeks Reference:

- Transition Table in Automata

---

3. Language of a Turing Machine

A language L is **recognized (accepted)** by a TM M if for all strings $w \in L$, M halts in an accepting state. If M halts, rejects, or loops indefinitely for $w \notin L$, the computation differs.

GeeksforGeeks Reference:

- Design Turing Machine for equal number of a's and b's
- Design TM for $L = \{a^n\, b^n\, n \geq 1\}$

Video Resource:

- Design TM for $a^n b^n c^n$
- Design TM for $a^n b^n$
- TM for addition/subtraction

---

4. Design and Types of Turing Machines

- **Single Tape TM:** Basic model
- **Multi-Tape TM:** Several tapes, each with a head; equivalent in power to single-tape, but usually more efficient.
- **Multi-Head TM:** Multiple heads move independently
- **Two-way Infinite Tape TM:** Tape infinite in both directions
- **Offline TM:** Input tape is read-only
- **Non-deterministic TM:** Simultaneous computation branches

Video Resource: Variants of Turing Machine
Watch here

GeeksforGeeks Reference:

- Universal Turing Machine
- Types of TM/variants
- Linear Bounded Automata

---

5. Church's Thesis & Universal/Restricted TM

- **Church-Turing Thesis:** All "effectively calculable" functions can be computed by a TM; TMs model any algorithmic process[4].
- **Universal TM:** A single TM that can simulate any other TM when provided with its description and input[5].
- **Restricted TMs:**
    - **Linear Bounded Automata (LBA):** Tape size limited by input, accepts context-sensitive languages.
    - **Other Variations:** Multi-tape, non-deterministic, offline TMs.

Video Resource: Church's Thesis in Theory of Computation
Watch here

GeeksforGeeks Reference:

- Church's Thesis for Turing Machine

- Universal Turing Machine

---

6. Decidable & Undecidable Problems, Halting Problem

- **Decidable (Recursive):** TM halts and answers 'yes' or 'no' for all inputs; examples: regular language recognition, checking parity.
- **Undecidable:** No algorithm (TM) can always decide; may loop forever for some inputs. Classic example: the Halting Problem.
- **Halting Problem:** Determines if TM halts for specific input; proven undecidable.

Video Resource: Decidability & Undecidability
Watch here
Halting Problem explanation (Sudhakar Atchala)

GeeksforGeeks Reference:

- Decidable and Undecidable Problems
- Halting Problem in Theory of Computation
- Decidability and Undecidability

---

7. Post's Correspondence Problem (PCP) & Modified PCP

- **PCP:** Given a set of dominos with top/bottom strings, can we select a sequence with equal concatenated strings? PCP is undecidable; no general algorithm exists[6].
- **Modified PCP (MPCP):** Match must begin with first domino, allows reduction to Turing machine simulation.

Video Resource: Post Correspondence Problem
Watch here
PCP solved examples (Sudhakar Atchala)

GeeksforGeeks Reference:

- Post Correspondence Problem
- Modified PCP

---

8. Complexity Classes: P, NP, NP-Complete, NP-Hard

- **P:** Problems solvable by deterministic algorithms in polynomial time.
- **NP:** Solutions verifiable in polynomial time by a non-deterministic TM.
- **NP-Complete:** Hardest problems in NP; if one is in P, then all are.
- **NP-Hard:** At least as hard as NP-complete problems; may not be in NP.

GeeksforGeeks Reference:

- Proof that SAT is NP Complete
- Quiz: P/NP/NP-Complete
- NP Hard & NP Complete Classes (TutorialsPoint)

Video Resource: NP-hard and Complexity explanation
Watch Complexity Video

9. Practice Questions

1. Define a Turing Machine to recognize $L = \{ww | w \in \{0, 1\}\}$.
2. Explain the difference between Universal TM and Linear Bounded Automata.
3. Give an example of a decidable and an undecidable problem.
4. Construct a transition table and diagram for a TM that accepts strings with equal numbers of a's and b's.
5. Prove that the Halting Problem is undecidable.

---

10. References

[1] Sudhakar Atchala. (2018–2025). Theory of Computation & Automata Theory Video Series. YouTube.
[2] GeeksforGeeks. Turing Machine and Computability Articles. https://www.geeksforgeeks.org/theory-of-computation/
[3] Tutorialspoint. Automata Theory (Turing Machine). https://www.tutorialspoint.com/automata_theory/turing_machine_introduction.htm
[4] Church's Thesis. https://www.geeksforgeeks.org/theory-of-computation/churchs-thesis-for-turing-machine/
[5] Universal Turing Machine. https://www.geeksforgeeks.org/compiler-design/universal-turing-machine/
[6] Post Correspondence Problem. https://www.geeksforgeeks.org/theory-of-computation/post-correspondence-problem/

Key Sudhakar Atchala TM Video Playlist:

- TM Examples and Design Playlist
- Theory of Computation / FLAT Full Playlist

Last Updated: November 24, 2025

Suitable For: B.Tech/CSE TOC/FLAT exam prep, project work, concept review.