Theory of Computation

Based on lectures by Dr. Arpit Sharma
Notes taken by Rwik Dutta

These notes are not endorsed by the lecturers, and I have modified them (often significantly) after lectures. They are nowhere near accurate representations of what was actually lectured, and in particular, all errors are almost surely mine.[1]

Contents

1 Finite Automata **2**

[1]This is how Dexter Chua describes his lecture notes from Cambridge. I could not have described mine in any better way.

1 Finite Automata

Definition 1 (Deterministic Finite Automaton)**.** A collection $(Q, \Sigma, \delta, q_0, F)$ such that

1. Q is a finite set of *states*.
2. Σ is a finite *alphabet*.
3. $\delta : Q \times \Sigma \to Q$ is the *transition function*.
4. $q_0 \in Q$ is the start state.
5. $F \subseteq Q$ is the set of *accepting states*.

We will use automata to solve set membership problems, i.e., given a finite alphabet Σ and a finite language $L \subset \Sigma^*$, we need to find whether a given string $x \in L$.

Definition 2 (Language of Automaton)**.** $L(M)$ of an automaton M is the set of all strings x that are accepted by the automaton.

Definition 3 (Extended transition function)**.** Let $M(Q, \Sigma, \delta, q_0, F)$ be a finite automaton. The extended transition function is given by

$$\begin{aligned} \delta^* &: Q \to \Sigma^* \\ \delta^*(q, \epsilon) &= q \\ \delta^*(q, xa) &= \delta(\delta^*(q, x), a) \end{aligned}$$

for all $q \in Q, x \in \Sigma^*, a \in \Sigma$. $\epsilon \in \Sigma^*$ represents the empty string.

Hence, $x \in L(M)$ if $\delta^*(q_0, x) \in F$.

Definition 4 (Regular language)**.** A language L is a regular language if $\exists$ some finite automaton M such that $L(M) = L$.