

Theory of Computation, CSCI 438 spring 2022
Definition of Regular languages, pg. 40-44, Jan. 14

A language is a set of strings

Given an alphabet Σ , a language on Σ is some subset of Σ^*

We want to create a formal definition of what it means for a language to be regular. Do this in three steps:

1. Define what it means for a machine to accept a string.
2. Define what it means for a machine to recognize a language.
3. Make the definition of regular languages be those languages which are accepted by a DFA.

Goal 1: Define what it means for a DFA to accept a string.

Let δ^* be the application of δ multiple times.

The signature of δ^* is:

$$\delta^*: Q \times \Sigma^* \rightarrow Q$$

Given a DFA $M = (Q, \Sigma, \delta, q_0, F)$, M accepts string $w \in \Sigma^*$ iff $\delta^*(q_0, w) \in F$.
(Page 36 but is written out in the text)

Goal 2: Define what it means for a DFA to recognize a language.

Given an DFA $M = (Q, \Sigma, \delta, q_0, F)$, $\mathcal{L}(M)$ is the set of strings accepted by M .

Goal 3: Define what it means for a language to be regular.

A language is regular iff some finite automaton recognizes it.
(Definition 1.6, page 40)