

Übung 1: Sprachen und Automaten

Theoretische Informatik Sommersemester 2013

Markus Kaiser

April 30, 2013

- Mail: tutor@zfix.org
- Web: tutor.zfix.org

- Wann?
 - Dienstag 10:15-11:45 00.08.038
 - Dienstag 12:05-13:35 00.08.038
- Übungsablauf, Aufgabentypen
- Hausaufgaben
 - Abgabe am Montag 14h, **allein**
 - Rückgabe in der **richtigen** Übung
 - Notenbonus für 40% der Punkte, 40% in der zweiten Hälfte
- Klausur
 - Endterm: Mi 31.07. 11.30-14h
 - Wiederholung: Do 26.09. 11-13.30h

Aus der VL

- Automatentheorie
 - Rechner mit endlichem oder kellerartigem Speicher
- Grammatiken
 - Syntax von Programmiersprachen
- Berechenbarkeitstheorie
 - Untersuchung der Grenzen, was Rechner prinzipiell können
- Komplexitätstheorie
 - Untersuchung der Grenzen, was Rechner mit begrenzten Ressourcen können

Definition

- Ein **Alphabet** Σ ist eine endliche Menge.
- Ein **Wort** über Σ ist eine endliche Folge von Zeichen.
- Eine Teilmenge $L \subseteq \Sigma^*$ ist eine **formale Sprache**

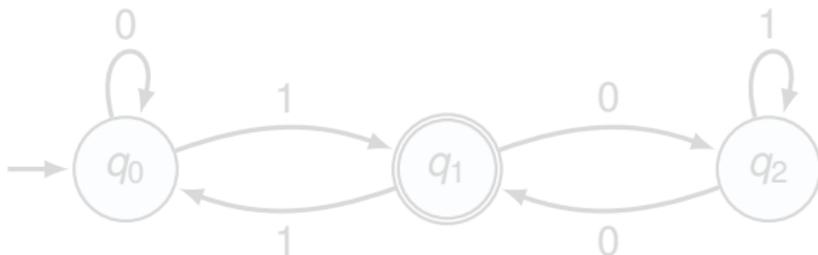
Definition (Operationen auf Sprachen)

- $AB = \{uv \mid u \in A \wedge v \in B\}$
- $A^n = \{w_1 \dots w_n \mid w_1 \dots w_n \in A\}, \quad A^0 = \{\epsilon\}$
- $A^* = \bigcup_{n \in \mathbb{N}_0} A^n$

Definition (Deterministischer endlicher Automat)

Ein **DFA** ist ein Tupel $M = (Q, \Sigma, \delta, q_0, F)$ aus einer/einem

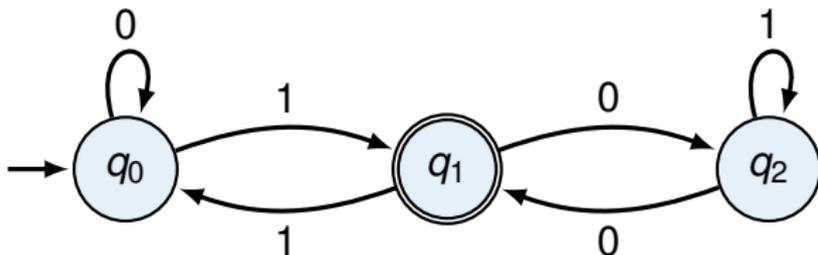
- endlichen Menge von **Zuständen** Q
- endlichen **Eingabealphabet** Σ
- totalen **Übergangsfunktion** $\delta : Q \times \Sigma \mapsto Q$
- **Startzustand** $q_0 \in Q$
- Menge von **Endzuständen** $F \subseteq Q$



Definition (Deterministischer endlicher Automat)

Ein **DFA** ist ein Tupel $M = (Q, \Sigma, \delta, q_0, F)$ aus einer/einem

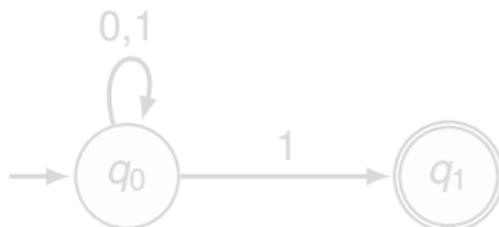
- endlichen Menge von **Zuständen** Q
- endlichen **Eingabealphabet** Σ
- totalen **Übergangsfunktion** $\delta : Q \times \Sigma \mapsto Q$
- **Startzustand** $q_0 \in Q$
- Menge von **Endzuständen** $F \subseteq Q$



Definition (Nicht-Deterministischer endlicher Automat)

Ein **NFA** ist ein Tupel $N = (Q, \Sigma, \delta, q_0, F)$ mit

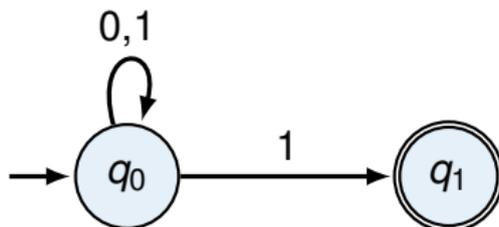
- Q, Σ, q_0, F wie ein DFA
- **Übergangsfunktion** $\delta : Q \times \Sigma \mapsto P(Q)$



Definition (Nicht-Deterministischer endlicher Automat)

Ein **NFA** ist ein Tupel $N = (Q, \Sigma, \delta, q_0, F)$ mit

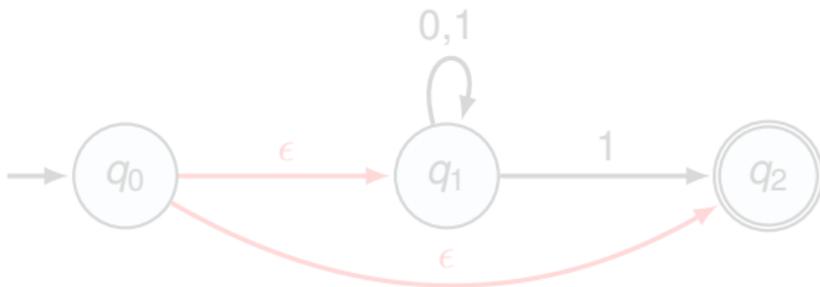
- Q, Σ, q_0, F wie ein DFA
- **Übergangsfunktion** $\delta : Q \times \Sigma \mapsto P(Q)$



Definition (NFA mit ϵ -Übergängen)

Ein ϵ -NFA ist ein Tupel $N = (Q, \Sigma, \delta, q_0, F)$ mit

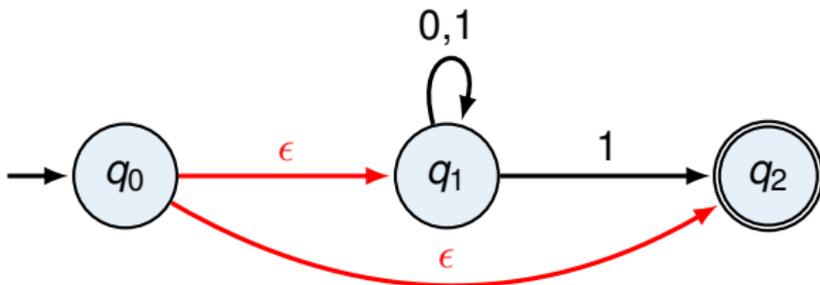
- Q, Σ, q_0, F wie ein DFA
- **Übergangsfunktion** $\delta : Q \times (\Sigma \cup \{\epsilon\}) \mapsto P(Q)$



Definition (NFA mit ϵ -Übergängen)

Ein ϵ -NFA ist ein Tupel $N = (Q, \Sigma, \delta, q_0, F)$ mit

- Q, Σ, q_0, F wie ein DFA
- **Übergangsfunktion** $\delta : Q \times (\Sigma \cup \{\epsilon\}) \mapsto P(Q)$



Übergangsfunktionen

Die Automaten $A = (Q, \Sigma, \delta, q_0, F)$ unterscheiden sich nur durch ihre Übergangsfunktionen.

$$\text{DFA } \delta : Q \times \Sigma \mapsto Q$$

$$\text{NFA } \delta : Q \times \Sigma \mapsto P(Q)$$

$$\epsilon\text{-NFA } \delta : Q \times (\Sigma \cup \{\epsilon\}) \mapsto P(Q)$$

Satz

DFA, NFA und ϵ -NFA sind gleich mächtig und lassen sich ineinander umwandeln.

