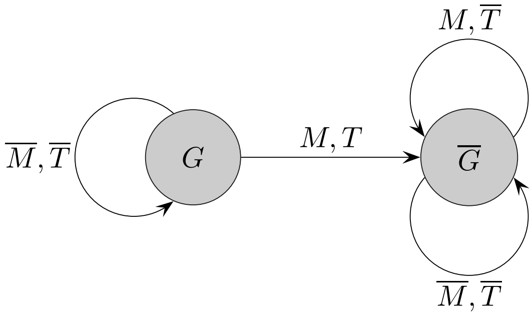
Endliche Automaten

Ein endlicher Automat verfügt über verschiedene interne Zustände. Ein Eingabewort wird Zeichen für Zeichen gelesen. Jeweils in Abhängigkeit vom letzten Eingabezeichen und dem momentanen Zustand geht der Automat in einen Folgezustand über, nachdem er ein Ausgabezeichen produziert hat.
Ein Zustand ist als Anfangszustand gekennzeichnet. Die ausgegebenen Zeichen bilden das Ausgabewort.
Ein endlicher Automat kann durch einen gerichteten Graphen beschrieben werden.
Als einfaches Beispiel nehmen wir eine Mausefalle.

Eingabealphabet	$E = \{M, \overline{M}\}$
mit der Bedeutung	$M =$ „Eine Maus kommt.“
	$\overline{M} =$ „Es kommt keine Maus.“
Zustandsmenge	$Z = \{G, \overline{G}\}$
mit der Bedeutung	$G =$ „Die Falle ist gespannt.“
	$\overline{G} =$ „Die Falle ist nicht gespannt.“
Ausgabealphabet	$A = \{T, \overline{T}\}$
mit der Bedeutung	$T =$ „Die Maus wurde getötet.“
	$\overline{T} =$ „Es wird keine Maus getötet.“



Ein Spezialfall eines endlichen Automaten ist ein Automat ohne Ausgabe (Akzeptor). Der Zweck besteht darin, eine Eingabe zu prüfen. Diejenigen Worte, die einen Akzeptor vom Anfangs- in den Endzustand überführen, bilden die Sprache des Akzeptors. Mit ihm können korrekt gebildete Zeichenreihen von anderen unterschieden werden. Das ist bei der Kompilierung eines Programms der Fall, denn vor der Erzeugung des Maschinencodes wird das Programm auf syntaktische Fehler untersucht. Diese Aufgabe übernimmt der Parser (engl. to parse grammatisch zerlegen), ein Bestandteil des Compilers.

1. Ein endlicher Automat ist gesucht, der alle 0-1-Wörter akzeptiert, die
   a) eine ungerade Anzahl Einsen enthalten,
   b) aus jeweils geradzahlig vielen Nullen und Einsen bestehen.

2. Ein endlicher Automat mit dem Eingabealphabet $\{a, b\}$ ist gesucht, der die Sprache
   a) $L = \{(abb)^n,\ n = 1, 2, ...\}$
   b) $L = \{a^{2n}b,\ n = 0, 1, 2, ...\}$ akzeptiert.

© Roolfs