

Verständnis formale Sprachen

Beantworten Sie kurz, präzise und mit Begründung folgende Fragen: (Die Begründungen müssen keine formellen mathematischen Beweise sein).

- (a) Welche Möglichkeiten gibt es, eine formale Sprache vom Typ 3 zu definieren?

- reguläre Grammatik
- endlicher Automat
- regulärer Ausdruck

- (b) Was ist die Komplexität des Wortproblems für Typ-3 Sprachen und wieso ist das so?

P , CYK-Algorithmus löst es in Polynomialzeit.

- (c) Sind Syntaxbäume zu einer Grammatik immer eindeutig? Falls nicht, geben Sie ein Gegenbeispiel.

Nein. Syntaxbäume zu einer Grammatik sind nicht immer eindeutig.

Gegenbeispiel

$G = (\{S, A, B\}, \{a\}, P, S)$

$P = \{$

$S \rightarrow AA$

$S \rightarrow BB$

$A \rightarrow a$

$B \rightarrow a$

$\}$

$S \vdash AA \vdash aA \vdash aa$

$S \vdash BB \vdash aB \vdash aa$

- (d) Wie kann man die Äquivalenz zweier Typ-3 Sprachen nachweisen?

Wir können von den auf Äquivalenz zu überprüfenden Sprachen jeweils einen minimalen endlichen Automaten bilden. Sind diese entstanden zwei Automaten äquivalent, so sind auch die Sprachen äquivalent.

- (e) Wie kann man das Wortproblem für das Komplement einer Typ-3 Sprache lösen?

Da das Komplement einer regulären Sprache wieder eine reguläre Sprache ergibt, kann das Wortproblem beim Komplement durch einen deterministisch endlichen Automaten gelöst werden.

- (f) Weshalb gilt das Pumping-Lemma für Typ 3 Sprachen?
- (g) Ist der Nachweis, dass das Typ-3 Pumping-Lemma für eine gegebene Sprache gilt, ausreichend, um zu zeigen, dass die Sprache vom Typ 3 ist? Falls nicht, geben Sie ein Gegenbeispiel, mit Begründung.
- (h) Geben Sie ein Beispiel, an dem deutlich wird, dass deterministische und nichtdeterministische Typ-2 Sprachen unterschiedlich sind.

Deterministisch Kontextfrei $L = \{ 0^n 1^n \mid n \geq 0 \}$

Nichtdeterministisch Kontextfrei $L = \{ \omega \omega^R \mid \omega \in \{0,1\}^* \}$ (R steht für rückwärts)

^a

^a<https://docplayer.org/19566652-Einfuehrung-in-die-theoretische-informatik.html>

- (i) Worin macht sich der Unterschied zwischen Typ 0 und 1 bemerkbar, wenn man Turingmaschinen benutzt, um das Wortproblem vom Typ 0 oder 1 zu lösen. Warum ist das so?

Typ 0: semi-entscheidbar, Typ 1: entscheidbar

Da Typ 1 nur Wörter verlängert, kann daher in Polynomialzeit überprüft werden, ob das Wort in der Sprache liegt, indem die Regeln angewendet werden, bis das Wortende erreicht ist.