

Übungen zur Vorlesung  
Automaten und Formale Sprachen  
Aufgabenblatt 6

Abgabe der Ausarbeitungen: 21.7.2006, spätestens 12.00 Uhr

Wo? Fächer beschriftet mit "Automaten und Formale Sprachen"  
in der Mitte der vierten Etage vor H426

**Aufgabe 21 (kontextfrei, nicht-kontextfrei)**(2+2+2+2+4+2 Punkte)  
Welche der folgenden Sprachen ist kontextfrei und welche nicht. Begründe!

1.  $L = \{a^n b^m \mid n \neq m\}$
2.  $L = \{a^i b^j c^k \mid i \leq j, j \leq k\}$
3.  $L = \{a^m b^n c^{m+n} \mid m, n \geq 0\}$
4.  $L = \{a^i b^j c^k \mid i \leq j, j \geq k\}$
5.  $L = \{a^m b^n c^{m+n-k} d^k \mid m, n, k \geq 0\}$
6.  $L = \{w \in \{a, b, c\}^* \mid w \text{ enthält gleichviele } a\text{'s, } b\text{'s und } c\text{'s}\}$

**Aufgabe 22 (Operationen auf kfS)**(3+4 Punkte)

1. Zeige, dass die Klasse der kontextfreien Sprachen nicht unter symmetrischer Differenz ( $A\Delta B = (A \setminus B) \cup (B \setminus A)$ ) abgeschlossen ist.  
(Tipp: Simuliere die Komplementoperation durch  $\Delta$ .)
2. Sei  $Perm(L) = \{w \mid \exists v \in L \text{ und Permutation } \pi \text{ mit } w = \pi(v)\}$ .  $w \in Perm(L)$ , wenn  $w$  durch vertauschen der Buchstaben eines  $v \in L$  entsteht. Zeige, dass die kontextfreien Sprachen unter  $Perm$  nicht abgeschlossen sind.  
(Tipp: Finde eine Sprache  $L$ , so dass  $Perm(L) = \tilde{L}$ , wobei  $\tilde{L}$  eine der Sprachen aus Aufgabe 21 ist und notwendigerweise nicht kontextfrei.)

**Aufgabe 23 (Parser)**(3+5+4+2 Punkte)

1. Erzeugen Sie aus der folgenden Grammatik mit dem in der Vorlesung vorgestellten Algorithmus einen Linksparser (Top-Down-Parser).  
 $G = (\{0, 1, *, +, (\cdot)\}, \{S, A, B\}, P, S)$   
 $P = \{S \rightarrow A, S \rightarrow A+A, A \rightarrow B, A \rightarrow B*B, B \rightarrow 0, B \rightarrow 1, B \rightarrow (S)\}$
2. Erzeugen Sie für  $G$  einen Rechtsparser (Bottom-Up-Parser).
3. Begründen sie warum der Rechtsparser einen Lookahead von Länge 1 braucht und dieser reicht um für alle  $w \in L(G)$  eine Ableitung anzugeben. (Zur Erklärung: Der Parser habe die Konfiguration  $(z, a_1 \dots a_k w, RB)$  und es gebe nun mehrere Ableitungsregeln in dieser Situation ( $B$  ist hier das oberste Kellerzeichen, auf das eine Regel angewendet werden soll). Der Parser hat Lookahead  $k$  wenn er durch Kenntnis der  $a_1 \dots a_k$  die 'richtige' Ableitungsregel determinieren kann. Die Ableitungsregel ist in diesem Sinne 'richtig', dass, falls das Wort in der Sprache ist, der Rest des Wortes komplett abgeleitet werden kann, so dass am Ende nur  $S$  noch auf dem Keller liegt.)
4. Verfolgt der CYK-Algorithmus einen Top-Down- oder Bottom-Up-Ansatz?

**Aufgabe 24 ( $LL(k)$ )**(3+3+6 Punkte)

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

$$P = \{S \rightarrow bBbA, S \rightarrow aBaA, A \rightarrow a, A \rightarrow aA, B \rightarrow a, B \rightarrow ab\}$$

1. Geben sie einen Linksparser für  $G$  an.
2. Die Sprache erzeugt vier verschiedene Wortklassen der Form  $q(a)^*$ , wobei  $q$  das Präfix sein soll. Geben sie diese an.
3. Zeige, dass der Parser einen Lookahead von 3 braucht und 2 zu wenig ist,  $G$  also eine  $LL(3)$  Grammatik ist. (Erklärung: Ist der Parser im Zustand  $(f, abba, Bba)$ , so bedeutet in diesem Fall Lookahead 3, dass der Parser wissen muss, dass der Rest der Eingabe mit  $aab$  anfängt, um die richtige Ableitungsregel zu wählen. In diesem Fall wäre  $B \rightarrow ab$  die richtige Wahl.)

**Aufgabe 25 (kontextsensitive Grammatik)**(6 Punkte)

$L = \{ww \mid w \in \{a, b\}^*\}$  ist nicht kontextfrei aber kontextsensitiv. Geben sie eine kontextsensitive Grammatik für  $L$  an.

(Tipp: Benutzen sie pro Terminal als erstes Regeln der Form  $S \rightarrow cC'S$ .  $C'$  soll Indikator sein, dass noch in der anderen Worthälfte an gleicher Stelle ein  $c$  zu setzen ist. Entwerfen sie nun Regeln, die es erlauben  $C'$  in die zweite Hälfte des Wortes zu schieben. Wandeln sie dann die  $C'$  in die entsprechenden Terminale um.)