Universität Karlsruhe

Informatik 1 WS 00/01

Institut für Telematik, Forschungsgruppe C&M
Prof. Dr. S. Abeck, R. Scholderer

Musterlösungen zum Übungsblatt 5

Lösung

1. Grammatik-Entwicklung (2 Punkte)

a)

$N = \{S, A\}$
$\Sigma = \{(,)\}$
$P = \{S \rightarrow A$
$A \rightarrow (A) \mid \varepsilon \}$

Eine CH-2 Grammatik, die die Sprache $(^n)^n$ für ein festes n erzeugt lautet:

$N = \{S, A, B\}$
$\Sigma = \{(,)\}$
$P = \{ S \rightarrow A) \mid \varepsilon$
$A \rightarrow (AB \mid ($
$B \rightarrow) \}$

Bei CH-2 Grammatiken ist die rechte Seite der Produktionen aus V^*. Das bedeutet, daß auch Produktionen der Form $A \rightarrow xAy$ ($x,y \in \Sigma^*$, $A \in V^*$)zugelassen sind, mit denen eine unendlich lange. gleichzeitige Vermehrung der Nichtterminale nach rechts und links möglich ist.

CH-3 Grammatiken können höchstens Produktionen der Form $A \rightarrow Ax$ bzw. $A \rightarrow xA$ ($x \in \Sigma$, $A \in N$) haben. Dies bedeutet, daß nur eine endlich lange Reihe von öffnenden Klammern erzeugt werden kann. Denn die Produktionen einer CH-3 Grammatik lassen es nicht zu, dass links und rechts von einem Nonterminal Terminale stehen, was für korrekte Klammerfolgen notwendig ist.
Bsp:
$G_{endl} = \{N, \Sigma, P, A\}$

$N = \{A, B, C, D, E, F\}$
$\Sigma = \{(,)\}$
$P = \{ A \rightarrow (B \mid (F$

$B \rightarrow (C \mid (E$
$C \rightarrow (D$
$D \rightarrow)E$
$E \rightarrow)F$
$F \rightarrow) \}$

Erzeugte Sprache:
$L(G_{endl}) = \{ (^n)^n, 1 \leq n \leq 3 \}$

b)
Zur Lösung dieser Aufgabe muß die in der Vorlesung vorgestellte Beispiel-Grammatik erweitert werden. Folgende Ergänzungen sind vorzunehmen:

- Zeichenvorrat $\sum$ wird um die Terminale / und - erweitert
- Zu den Produktionen werden A $\rightarrow$ **A - T** und T $\rightarrow$ **T / F** hinzugefügt

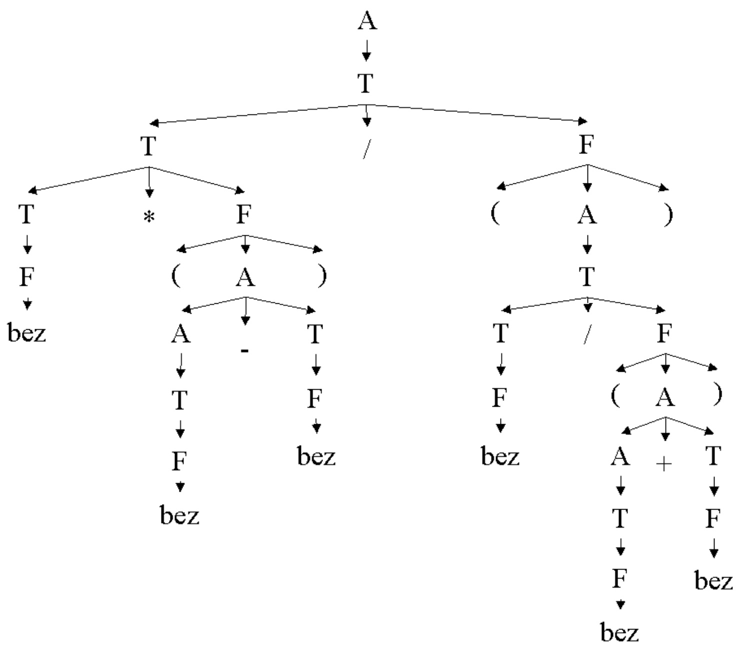
Grammatik $G_A = (\sum, N, P, A)$

N = {bez, -, /, +, *, (,) }
$\sum$ = A, T, F}
P = A $\rightarrow$ T | A + T | **A - T**,
T $\rightarrow$ F | T * F | **T / F**,
F $\rightarrow$ bez | (A) }

A ist das Startsymbol (Axiom, Ziel)

Backus Naur Form für die Grammatik:
<Ausdruck> ::= <Term> | <Ausdruck>[+|-] <Term>
<Term> ::= <Faktor> | <Term>[*|/] <Faktor>
<Faktor> ::= bez | (<Ausdruck>)

Ableitungsbaum: bez*(bez-bez)/(bez/(bez+bez))



Lösung

2. Backus Naur Form (1,5 Punkte)

<pziffer> ::= 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
<ziffer> ::= 0 | <pziffer>
<zahl> ::= <pziffer> {<ziffer>}*
<kommazahl> ::= [0 | <zahl>] .{<ziffer>}* <pziffer>
<dezimal_zahl> ::= 0 | {-} [<zahl> | < kommazahl >]

Lösung

3. Graphen - Aufbau eines Graphen (1 Punkt)

a)

	0	1	2	3	4	5	6
0	0	0	0	0	0	0	1
1	0	0	0	0	1	1	0
2	0	1	0	0	1	0	0
3	0	0	1	1	0	0	0
4	0	0	1	1	1	1	0
5	0	1	0	0	1	0	1
6	1	0	0	0	0	0	0

Adjazenzmatrix zum Graphen aus Abb. 1

	0	1	2	3	4	5	6
0	0	0	0	1	0	0	1
1	0	0	0	0	1	1	0
2	0	1	0	0	1	0	0
3	0	1	1	1	0	0	0
4	0	0	1	1	1	1	0
5	0	1	0	0	1	0	1
6	1	0	0	0	0	0	0

Adjazenzmatrix zum Graphen aus Abb. 2

b)

Knoten	Eingangsgrad |·e|	Ausgangsgrad |e·|
0	1	1
1	2	2
2	2	2
3	2	2
4	4	4
5	2	3
6	2	1

Gesamtgrad des Graphen: grad(G) = 15

Lösung

4. Darstellung und Erreichbarkeit (3 Punkte)

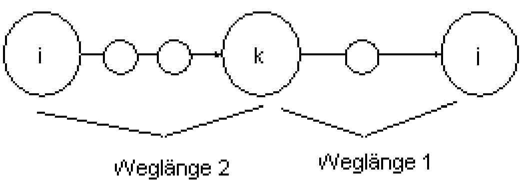
a. Adjazenzmatrix für den gegebenen Graphen. Zur besseren Lesbarkeit sind in der untenstehenden Abbildung die Nullen nicht aufgeführt.

nach / von	0	1	2	3	4	5	6	7	8	9
0			1					1	1	
1	1									
2		1		1						
3						1				
4			1							
5					1					
6								1		
7			1							
8										1
9							1			

b)
Wege von Endsystem1 zu Endsystem5:
Erster Weg: e1 -> e0 -> e2 -> e3 -> e5
Zweiter Weg: e1 -> e0 -> e7 -> e2 -> e3 -> e5
Dritter Weg: e1 -> e0 -> e8 -> e9 -> e6 -> e7 -> e2 -> e3 -> e5

c)
1. Die transitive Hülle eines Graphen enthält alle möglichen Verbindungen zwischen seinen Knoten.
2. Der Floyd-Warshall-Algorithmus berechnet die transitive Hülle eines Graphen.

d)
Vorbemerkung:
Es gilt für einen Weg von Knoten i über Knoten k nach Knoten j:



Kein Zwischenknoten ist größer als k-1.
Die Länge eines Weges ist definiert als die Anzahl der durchschrittenen Zwischenknoten.

A sei die Adjazenzmatrix des Graphen. $A\sigma_{(k)}$ ist die Adjazenzmatrix des Graphen der nach dem k-ten Schritt des Floyd-Warshall-Algorithmus entsteht. Neu hinzugekommene Verbindungen sind jeweils fett gedruckt.

Initialisierungsschritt:

$\sigma^{(init)}$ Der Algorithmus wird über die Adjazenzmatrix und die Einheitsmatrix initialisiert:

$A\sigma_{(init)} = A + E =$

	0	1	2	3	4	5	6	7	8	9
0	1		1					1	1	
1	1	1								
2		1	1	1						
3				1		1				
4			1		1					
5					1	1				
6							1	1		
7			1					1		
8									1	1
9							1			1

k =0: $\sigma^{(0)} = \sigma^{(init)} \cup \{(i,j) \mid i \, \sigma^{(init)} \, 0 \text{ und } 0 \, \sigma^{(init)} \, j\}$ "Wege über den Knoten Null"

$A\sigma_{(0)} =$

	0	1	2	3	4	5	6	7	8	9
0	1		1					1	1	
1	1	1	**1**					**1**	**1**	
2		1	1	1						
3				1		1				
4			1		1					
5					1	1				
6							1	1		
7			1					1		
8									1	1
9							1			1

k =1: $\sigma^{(1)} = \sigma^{(0)} \cup \{(i,j) \mid i \, \sigma^{(0)} \, 1 \text{ und } 1 \, \sigma^{(0)} \, j\}$ zugelassene Zwischenknoten = {0,1}

$A\sigma_{(1)} =$

	0	1	2	3	4	5	6	7	8	9
0	1		1					1	1	
1	1	1	1					1	1	
2	**1**	1	1	1				**1**	**1**	
3				1		1				
4			1		1					
5					1	1				
6							1	1		
7			1					1		
8									1	1
9							1			1

k = 2: $\sigma^{(2)} = \sigma^{(1)} \cup \{(i,j) \mid i \, \sigma^{(1)} \, 1 \text{ und } 1 \, \sigma^{(1)} \, j\}$ zugelassene Zwischenknoten = {0,1,2}

$A\sigma_{(2)} =$

	0	1	2	3	4	5	6	7	8	9
0	1	**1**	1	**1**				1	1	
1	1	1	1	**1**				1	1	
2	1	1	1	1				1	1	
3				1		1				
4	**1**	**1**	1	**1**	1			**1**	**1**	
5					1	1				
6							1	1		
7	**1**	**1**	1	**1**				1	**1**	
8									1	1
9							1			1

<u>k = 3:</u> $\sigma^{(2)} = \sigma^{(2)} \cup \{(i,j) \mid i\ \sigma^{(2)}\ 1 \text{ und } 1\ \sigma^{(2)}\ j\}$ zugelassene Zwischenknoten = {0,1,2,3}

$A\sigma_{(3)} =$

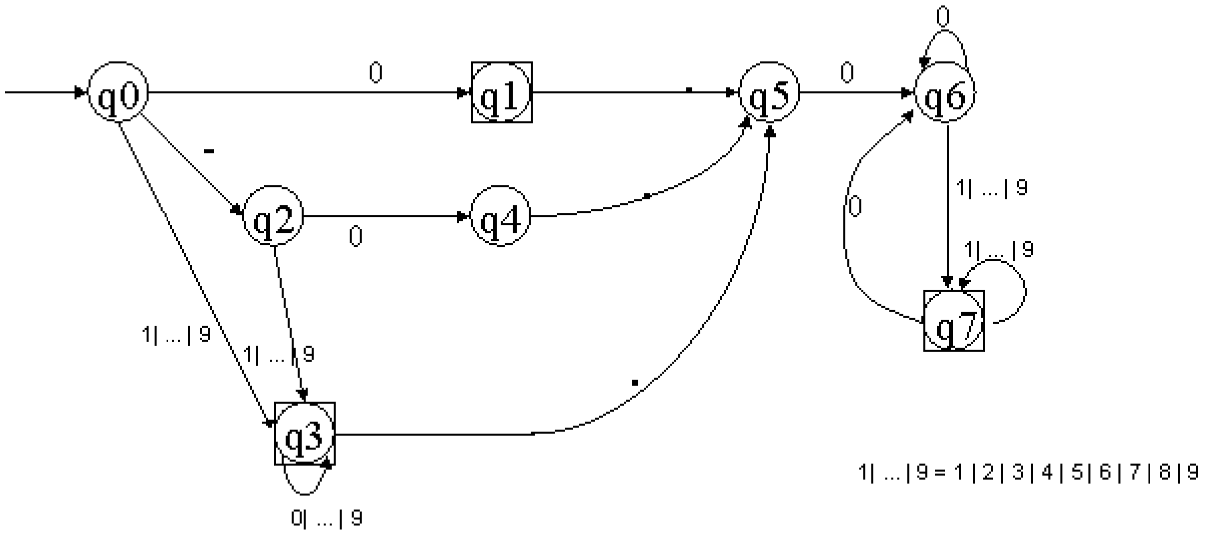
	0	1	2	3	4	5	6	7	8	9
0	1	1	1	1		**1**		1	1	
1	1	1	1	1		**1**		1	1	
2	1	1	1	1		**1**		1	1	
3				1		1				
4	1	1	1	1	1	**1**		1	1	
5					1	1				
6							1	1		
7	1	1	1	1		**1**		1	1	
8									1	1
9							1			1

e)
Von e0 aus sind in dem neuen Graphen die Knoten e1, e2, e3, e5, e7 und e8 erreichbar.
Da $A\sigma_{(3)}$ an der Stelle (1,5) den Eintrag 1 hat, existiert nach drei Durchläufen ein Weg von e1 nach e5.
Da $A\sigma_{(2)}$ an der Stelle (0,9) den Eintrag 0 hat, existiert nach zwei Durchläufen kein Weg von e0 nach e9.

Lösung

5. EA-Automaten für reelle Zahlen (1,5 Punkte)

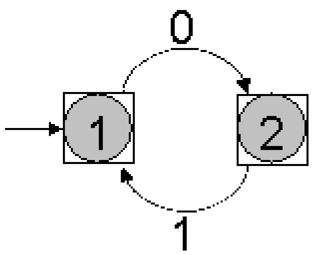
Σ = {1, 2, 3, 4, 5, 6, 7, 8, 9, 0, -, .}
Q = {q1, q2, q3, q4, q5, q6, q7}
Anfangszustand: q0
F = {q1, q3, q7}
Darstellung der Zustandsübergänge als Graph:



Lösung

6. Java: Endlicher Automat (1 Punkt)

Der in der Aufgabenstellung abgebildete Automat muss noch durch Endzustände ergänzt werden.



Das folgende Programm zeigt die Funktionsweise dieses Automaten.

```
import info1.*;
class Automat {
        public static void main( String[] args ) {
                int a;                                                      // Vari
                int zustand   = 1;                                  // zustand beschreibt d

                System.out.println("Geben Sie eine Zahl ein ");

                boolean korrekt = true;                             // Initialisierung der 
                while (korrekt == true) {                           // Beginn der Schleife
                        a = Console.in.readInt();                   // Einlesen der Zahl
                        if (a == 0 && zustand == 1) {
                                zustand = 2;
                                System.out.println("Zahl akzeptiert");

                        } else {
```

```
                                if (a == 1 && zustand == 2) {
                                        zustand = 1;
                                        System.out.println("Zahl akzeptiert");

                                } else {
                                        if (!(a == 1)&& !(a == 2)) {
                                                System.out.println("Wort zu Ende.");
                                        } else {
                                                System.out.println("Zahl nicht akzeptie
                                        }
                                korrekt = false;
                                }
                        }
                }
        }
}
```

Bemerkung:
Jeder endliche Automat kann eine CH-3 Sprache akzeptieren. Der Automat aus dieser Aufgabe befindet sich genau dann in einem Endzustand, wenn die Eingabe ein mit der folgenden Grammatik erzeugtes Wort war.

$S \rightarrow \varepsilon \mid 0 \mid 0E$

$E \rightarrow 1 \mid 1N$

$N \rightarrow 0 \mid 0E$
