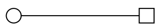
Prof. Dr. Hans-Jörg Kreowski
Studiengang Informatik
Universität Bremen

28. April 2004

Formale Sprachen: Graphentransformation

1. Übungsblatt

Es werden in diesem Übungsblatt nur ungerichtete und unmarkierte Graphen behandelt. Es gibt also nur eine unsichtbare Markierung, die alle Kanten tragen und deshalb keine Rolle spielt. Und die Kanten treten paarweise auf, die jeweils entgegengesetzt laufen. In der visuellen Darstellung kann man das Paar gerichteter Kanten durch eine ungerichtete Kante ersetzen:

 kurz für 

Ensprechend werden als Zahl der Kanten die Paare gezählt. Die Menge dieser Graphen wird mit $\mathcal{G}$ bezeichnet. $\mathcal{G}_0$ ist die Teilmenge aller Graphen ohne Schleifen und Mehrfachkanten (d.h. zwischen je zwei Knoten gibt es höchstens eine ungerichtete Kante). Als Regeln stehen alle zur Verfügung, deren Graphen höchstens drei Knoten und drei Kanten enthalten. Also

ist eine zulässige Regel. Diese Graphen stehen auch als Startgraphen zur Verfügung.

1. Es soll eine möglichst interessante Graphsprache $L \subseteq \mathcal{G}$ generiert werden. Dazu soll ein Startgraph Z und eine Menge von Regeln P gewählt werden, so dass $L = L(P, Z) = \{M \in \mathcal{G} \mid Z \overset{*}{\underset{P}{\Longrightarrow}} M\}$.

   Die Sprache L soll außer durch P und Z noch mindestens umgangssprachlich beschrieben werden, möglichst aber durch eine graphentheoretische Eigenschaft o.ä.

   Wählt man beispielsweise als P_2 die Menge, die genau die obige Regel enthält, und ihre linke Seite als Z_2, dann ist $L(P_2, Z_2)$ gerade die Menge aller 2-Bäume L_2. 2-Bäume werden üblicherweise rekursiv definiert als die kleinste Menge von Graphen, die Z_2 enthält und bei der jeder andere 2-Baum dadurch aus einem 2-Baum entsteht, dass man noch einen Knoten hinzunimmt und diesen durch zwei neue Kanten mit Quelle und Ziel einer vorhandenen Kante verbindet.

   Solange euch nichts anderes einfällt, könnt ihr versuchen, folgende Sprachen in der genannten Art zu generieren:

   (1) die Menge aller Graphen $\mathcal{G}$,

   (2) die Menge aller diskreten Graphen $\mathcal{D}$ (deren Kantenmenge leer ist),

   (3) die Menge aller Bäume $\mathcal{B}$,

   (4) die Menge aller einfachen Kreise $\mathcal{C}$,

   Statt einer Sprache dürfen auch mehrere Beispiele generiert werden.

2. Wenn in der ersten Aufgabe L unabhängig von P und Z formalisiert ist, kann man versuchen, $L = L(P, Z)$ zu beweisen. Gilt beispielsweise $Z \overset{n}{\underset{P}{\Longrightarrow}} M$ impliziert $M \in L$ für alle $n \in \mathbb{N}$, was sich oft durch Induktion über n ergibt, so folgt daraus bereits $L(P, Z) \subseteq L$. Die Umkehrung $L \subseteq L(P, Z)$ ist meist schwerer zu zeigen.

   Betrachte z.B. $Z_2 \overset{n}{\underset{P_2}{\Longrightarrow}} M$. Für $n = 0$ als Induktionsanfang ergibt sich $M = Z_2 \in L_2$ nach Definition. Und im Induktionsschluss kann man $Z_2 \overset{n+1}{\underset{P_2}{\Longrightarrow}} M$ in $Z_2 \overset{n}{\underset{P_2}{\Longrightarrow}} N$ und $N \underset{P_2}{\Longrightarrow} M$ zerlegen. $N \in L_2$ folgt dann aus der Induktionsvoraussetzung, und M entsteht aus N, wie in der Definition von L_2 beschrieben. Also ist auch M ein 2-Baum.

   Umgekehrt kann man im Falle von 2-Bäumen Induktion über die Zahl der Knoten machen.

   IA: Z_2 ist einziger 2-Baum mit 2 Knoten, und es gilt: $Z_2 \overset{0}{\underset{P_2}{\Longrightarrow}} Z_2$. Also: $Z_2 \in L(P_2, Z_2)$.

   IV: $M \in L_2$ mit $\#M = n \geq 2$ impliziert $M \in L(P_2, Z_2)$.

   IS: Betrachte $M \in L_2$ mit $\#M = n + 1$. Nach Definition von M existiert $N \in L_2$ mit $N \underset{P_2}{\Longrightarrow} M$ und $\#N = n$. Nach IV gilt $N \in L(P_2, Z_2)$, was $Z_2 \overset{*}{\underset{P_2}{\Longrightarrow}} N$ bedeutet. Zusammen erhält man eine Ableitung $Z_2 \overset{*}{\underset{P_2}{\Longrightarrow}} N \underset{P_2}{\Longrightarrow} M$ und damit $M \in L(P_2, Z_2)$.

3. Statt Sprachen zu generieren, kann man sie auch durch einen umgekehrten Prozess, der manchmal Reduktion genannt wird, erkennen:

$$L = R(P, Z) = \{M \in \mathcal{G} \mid M \underset{P_2}{\Longrightarrow} Z\}.$$

   Statt $\mathcal{G}$ darf auch mit den Graphen aus $\mathcal{G}_0$ begonnen werden. Die resultierende Sprache wird dann mit $R_0(P, Z)$ bezeichnet.

   Wieder sollen möglichst originelle Sprachen auf diese Weise modelliert werden.

4. Wieder soll möglichst bewiesen werden, dass L und $R(P, Z)$ für eine bestimmte Wahl von L und P und Z übereinstimmen (bzw. L und $R_0(P, Z)$).

   Im Falle der 2-Bäume ist leicht zu sehen, dass $L(P_2, Z_2) = L_2 = R_0(P_2^{-1}, Z_2)$ gilt, wenn P^{-1} für eine Regelmenge P die Regeln enthält, die aus P entstehen, indem man linke und rechte Seiten vertauscht.

Zur Bewertung:

In 1 und 3 kann man je 30 Punkte erreichen, in 2 und 4 je 20. Die vier expliziten Beispiele in 1 ergeben je 5 Punkte, eigene Beispiele sind 5 bis 10 Punkte wert. Dasselbe gilt für 3. In 2 sollen zwei Inklusionen der Form $L \subseteq L(P, Z)$ oder $L(P, Z) \subseteq L$ bewiesen werden. (Sie dürfen allerdings nicht trivial sein wie $L(P, Z) \subseteq \mathcal{G}$.) Dasselbe gilt für 4. Da 1 und 3 sowie 2 und 4 sehr ähnlich sind, kann weniger im einen durch mehr im anderen ausgeglichen werden.

Außerdem werden die drei originellsten Beispiele mit einem Buchpreis prämiert.

Laufzeit bis spätestens 25. Mai 2004. Zwischenergebnisse können jederzeit vorher abgegeben werden.