Universität Karlsruhe (TH)

Institut für Programmstrukturen und Datenorganisation (IPD)

Informatik I WS 2003/04 http://www.infoeins.de

Dozent: Prof. Dr.rer.nat. G. Goos goos@ipd.info.uni-karlsruhe.de

Übungsleiter: Tom Gelhausen gelhausen@fzi.de

Musterlösung 3 - (60T / 0P)

Grammatiken & Ableitungsbäume

Ausgabe: 31.10.2003

Abgabe: 07.11.2003 13:30 Uhr

Einwurf im Keller des Informatik-Hauptbaus (Geb. 50.34)

1 Algorithmenbegriff (11T)

In der Vorlesung wurde der „Markov-Algorithmus“ eingeführt. Untersuchen Sie, ob es sich der Definition nach wirklich um einen Algorithmus handelt. Gehen Sie dabei auf folgende Punkte ein:

- Eingabe & Ausgabe (1T)
- Operationen (Beschreibung!) (4T)
- Eigenschaften (Begründung!) (4T)
- Turingmächtigkeit (1T)

Inwiefern hat der Markov-Algorithmus Vorteile gegenüber dem Semi-Thue-System (1T)?

Lösung:

Eingabe: Wort x

Operationen: priorisierte Ersetzungsregeln,
- es wird immer die Regel mit der höchsten Priorität angewandt
- sie wird soweit als möglich links im Wort angewandt
- korrekte Terminierung mit Halteregeln bzw.
- fehlerhafte Terminierung wenn keine Regel mehr angewendet werden kann

Ausgabe: Ableitung von x

Eigenschaften:

- potentiell nicht terminierend (z.B. a $\Rightarrow$ a)
- deterministisch (es wird immer die höchstpriore Regel soweit links als möglich angewandt)
- determiniert (da deterministisch und Regeln auf der rechten Seite kein *oder* haben)
- sequentiell (es wird immer genau eine Regel nach der anderen angewandt)

Turing-Mächtigkeit:
Die Markov-Algorithmen mit Schiffchen erfüllen, wie oben gezeigt, den Algorithmenbegriff und sind wie die Semi-Thue-Systeme nach der Churchschen These turingmächtig.
Die Priorisierung der Regeln bei den Markov-Algorithmen führt zu einer Beschränkung der möglichen Regelanwendungen, was ihre Mächtigkeit einschränkt. Erst durch Einführung von Schiffchen wird diese Beschränkung wieder aufgehoben und Markov-Algorithmen turing-mächtig gemacht.

Markov-Algorithmen ermöglichen im Gegensatz zu Semi-Thue-Systemen einen deterministischen Ablauf und erlauben es zwischen erfolgreicher Beendigung und einem fehlerhaften Halt zu unterscheiden.

2. Chomsky-Grammatiken (27T)

2.1 Grammatik 1 (5T)

Gegeben sei folgende Grammatik:

G_1 = { Σ, N, P, S }
S = Formel
N = { Formel, Atom, Negation, KonDis, BinOp, Stern }
Σ = { P, ^{x}, ¬, ∧, ∨, (,) }

P = {	Formel	↦	Atom | Negation | KonDis,
	Atom	↦	Atom Stern | P,
	Stern	↦	^{x} Stern | ε,
	Negation	↦	¬ Formel,
	KonDis	↦	(Formel BinOp Formel),
	BinOp	↦	∨ | ∧ }

Geben Sie den einschränkensten Chomsky Grammatiktyp an (1T) und begründen Sie ihre Entscheidung (2T). Stellen Sie die erzeugte Sprache in erweiterter Backus Naur Form dar, verwenden Sie dabei so wenige Produktionen wie möglich (2T).

Lösung:
Die Grammatik ist vom Typ Chomsky 2.
Auf den linken Seiten der Produktionen kommt ausschließlich ein einzelnes Nichtterminal vor, also ist die Grammatik von Typ 2 oder Typ 3. Da aber auf der rechten Seite bei der KonDis-Produktion Terminale rechts und links von Nichtterminalen stehen, kann es nicht Typ 3 sein.

Sprache in EBNF:

⟨Formel⟩ ::= 'P' ('^{x}')* | '¬'⟨Formel⟩ | '(' ⟨Formel⟩ ('∨' | '∧') ⟨Formel⟩ ')'

Hinweis:
Die Grammatik beschreibt den Sprachschatz der aussagenlogischen Formeln. Sie ist ferner vom Typ SLL(1), der in Info 3 behandelt wird. Dies ist eine Einschränkung im Vergleich zu kontextfreien Sprachen, ermöglicht aber die Konstruktion eines deterministischen Algorithmus zur Akzeption der Sprache (Verwendung z.B. im Übersetzerbau).

Hinweis für Tutoren:
Ein 1:1-Umschreiben der Produktionen in BNF gibt nur 0,5T.

2.2 Grammatik 2 (5T)

Gegeben sei die folgende Grammatik:

$G_2 = \{ \Sigma, N, P, S \}$

$N = \{ S, A, N, U, Z \}$

$\Sigma = \{ s, a, n, u \}$

$P = \{ S \mapsto ASA \mid NSN \mid USU \mid ZSZ \mid a \mid n \mid s \mid u,$

$A \mapsto a,$

$N \mapsto n,$

$U \mapsto u,$

$Z \mapsto s \}$

Geben Sie den einschränkensten Chomsky Grammatiktyp an (1T) und begründen Sie ihre Entscheidung (1T). Beschreiben Sie ferner die Eigenschaft der erzeugten Worte (1T) und konstruieren Sie einen Ableitungsbaum für das Wort „annasusanna“ (2T).

Lösung:

Die Grammatik ist vom Typ Chomsky 2.

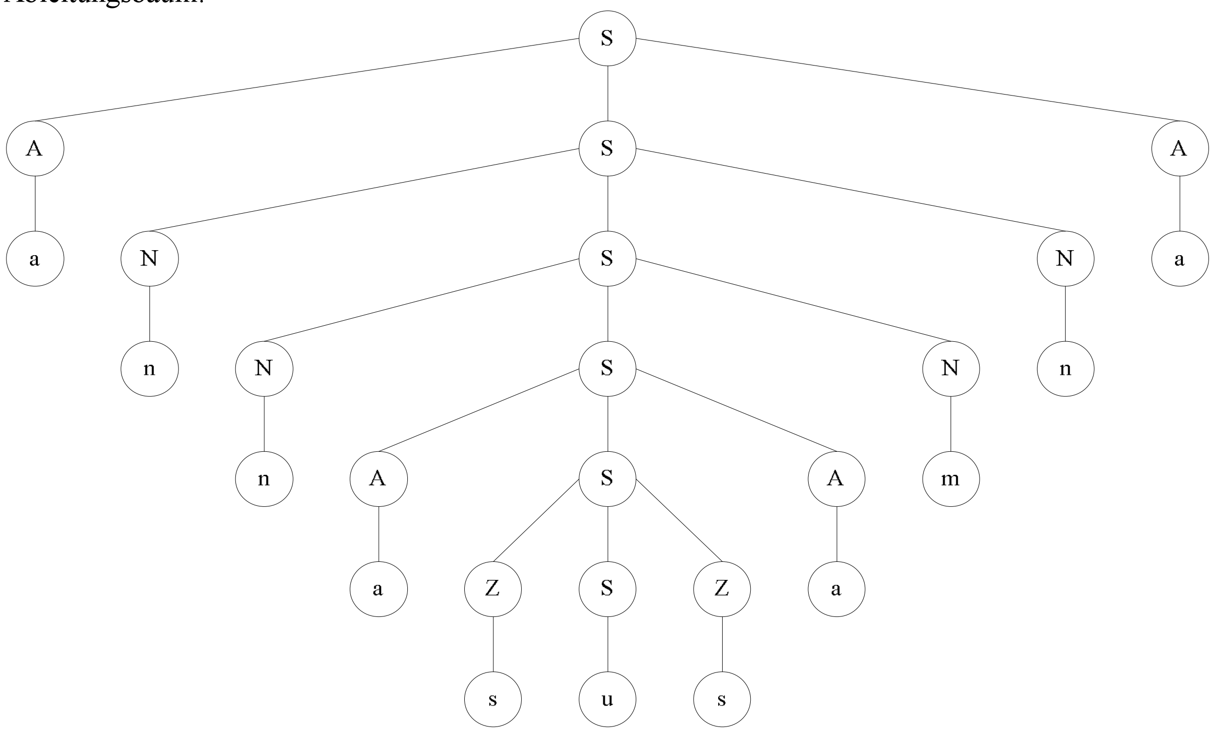
Auf den linken Seiten der Produktionen stehen ausschließlich einzelne Nichtterminalzeichen, also ist die Grammatik vom Typ 2 oder 3. Auf der rechten Seite kommen aber Ausdrücke wie ASA vor, was Typ 3 ausschließt.

Die erzeugten Worte sind Palindrome, d.h. sie sind von vorne und von hinten gelesen genau gleich.

Formal:

$L(G_2) = \{ wzw^R \mid w \in \Sigma^*, z \in \Sigma \}$ (w^R stellt hierbei ein umgedrehtes Wort dar, z.B. $(\text{hallo})^R = \text{ollah}$)

Ableitungsbaum:



2.3 Grammatik 3 (5T)

Gegeben sei die folgende Grammatik:

$G_3 = \{ \Sigma, N, P, S \}$

$N = \{ S, X, Y, Z \}$

$\Sigma = \{ b, e, h, l, n, t \}$

$$
\begin{aligned}
P = \{ \; & S \mapsto X, \\
& X \mapsto Ye \mid Yt \mid Yen, \\
& Y \mapsto hZ \mid lZ \mid bZ, \\
& Z \mapsto eb \; \}
\end{aligned}
$$

Geben Sie den einschränkensten Chomsky Grammatiktyp an (1T) und begründen Sie ihre Entscheidung (1T). Welche Sprache erzeugt diese Grammatik (2T) ? Unabhängig von der Grammatik, welches ist der einschränkenste Chomsky-Typ dieser Sprache (1T) ?

Lösung:

Die Grammatik ist vom Typ Chomsky 2.

Es treten sowohl links- als auch rechtslineare Produktionen auf somit kann der Typ nicht Chomsky 3 sein. Da aber auf der linken Seite nur einzelne Nichtterminale vorkommen, muss es sich um Chomsky 2 handeln.

$L(G_3)$ = { lebe, lebt, leben, bebe, bebt, beben, hebe, hebt, heben}

Die erzeugte Sprache liegt in Chomsky 3.

Hinweis: Sie kann durch einen *regulären Ausdruck* dargestellt werden (siehe Kapitel 2). Alle Sprachen, die derart dargestellt werden können sind vom Typ Chomsky 3.

2.4 Grammatik 4 (6T)

Geben Sie zur Sprache der arithmetischen Ausdrücke, in denen die Addition, Subtraktion, Multiplikation, Division als Operatoren auftreten, eine Grammatik an (2T).

Beschreiben Sie die Sprache in Backus-Naur-Form (1T).

Geben Sie zum Ausdruck **bez · (bez – bez) / (bez / (bez + bez))** den Ableitungsbaum an (3T).

Lösung:

$G_4 = \{ \Sigma, N, P, A \}$

$N = \{ A, T, F \}$

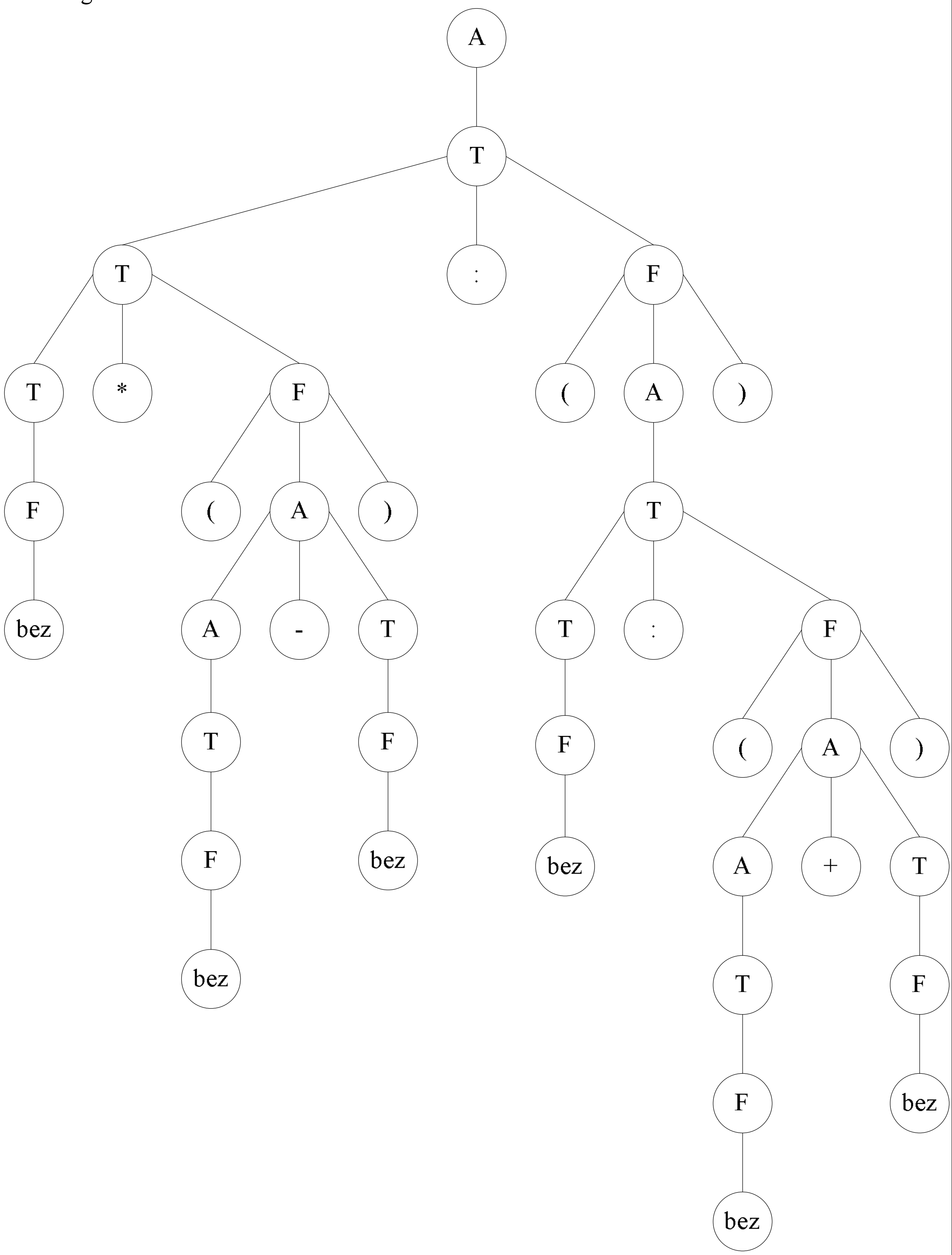
$\Sigma = \{ \text{bez}, -, :, +, \cdot, (,) \}$

$$
\begin{aligned}
P = \{ \; & A \mapsto T \mid A + T \mid A - T, \\
& T \mapsto F \mid T \cdot F \mid T : F, \\
& F \mapsto \text{bez} \mid (A) \; \}
\end{aligned}
$$

EBNF:

⟨Ausdruck⟩ ::= ⟨Ausdruck⟩ '+' ⟨Ausdruck⟩ | ⟨Ausdruck⟩ '*' ⟨Ausdruck⟩ | ⟨Ausdruck⟩ '-' ⟨Ausdruck⟩ **|** ⟨Ausdruck⟩ ':' ⟨Ausdruck⟩ **|** '('⟨Ausdruck⟩')' **|** 'bez'

Ableitungsbaum:



2.5. Grammatik-Transformation (2T)

Gegeben sei die folgende Grammatik vom letzten Übungsblatt:

$G_5 = \{ \Sigma, N, P, S \}$
$N = \{ S, I, N \}$
$\Sigma = \{ n, i \}$
$P = \{ S \mapsto N,$
$\quad I \mapsto iN \mid i,$
$\quad N \mapsto nI \}$

Wandeln Sie diese Grammatik in eine Grammatik vom Typ Chomsky 3, die immer noch die gleiche Sprache erzeugt.

Lösung:

$G_{5'} = \{ \Sigma, N, P, S \}$
$N = \{ S, I, N \}$
$\Sigma = \{ n, i \}$
$P = \{ S \mapsto nI,$
$\quad I \mapsto iN \mid i,$
$\quad N \mapsto nI \}$

2.6 Begriffsdefinitionen (4T)

Definieren Sie die Begriffe Syntax, Semantik, Chomsky-Grammatik und Sprache (je 1T).

Lösung:
Sei T die Menge der Terminalsymbole, dann ist T* die Menge aller Worte.

Syntax:
Menge der Regeln, die festlegt, welche Elemente aus T* zur Sprache gehören.

Semantik:
Bedeutung der Zeichenkette, z.B. „1234“ als Zahl

Chomsky-Grammatik:
Quadrupel $G = \{ \Sigma, N, P, S \}$ mit
N: Menge der Nichtterminale
Σ: Menge der Terminale
P: Menge der Produktionen (syntaktische Regeln)
S: Startsymbol $\in N$

Sprache:
Die durch eine Chomsky-Grammatik G definierte Sprache L ist die Menge:
$L(G) = \{ x \mid S \mapsto^* x \mid x \in T^* \}$

Aufgabe 3: Kantorowitsch- Bäume (10T)

3.1 Umwandeln von Formeln (5T)

Zeichnen Sie die Kantorowitsch- Bäume zu folgenden mathematischen Formeln
(*Beachte*: es gelten die üblichen mathematischen Vorrang-Regeln):

a. $3 \cdot a + 7 \cdot b$ (1T)

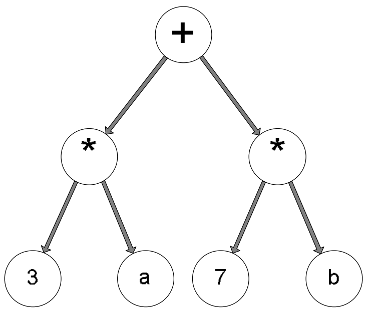
b. $(-12 + 11) \cdot 80$ (1T)

c. $((-5 \cdot x) + (-7 \cdot y)) / (25 + z)$ (1,5T)

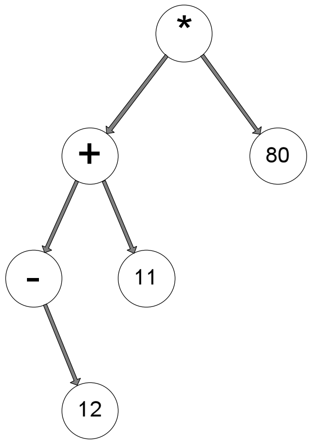
d. $((3 \cdot a + 2) \cdot a + 1) \cdot a$ (1,5T)

Lösung:

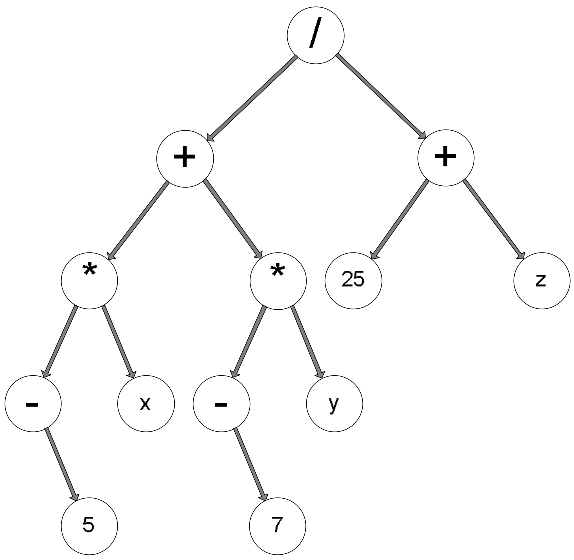
a.



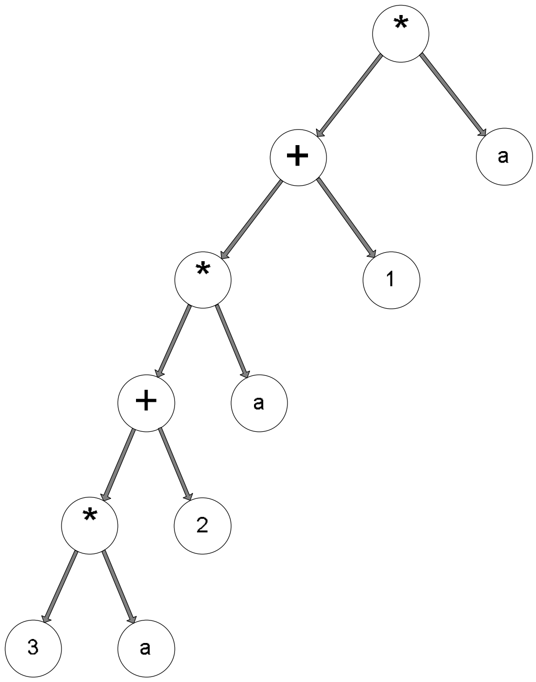
b.



c.



d.



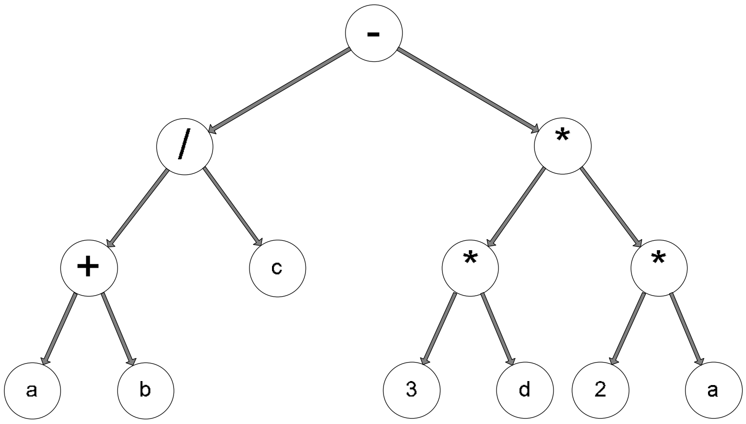
3.2 Infix-, Präfix-, Postfix-Darstellung (5T)

Zeichnen Sie den Kantorowitsch-Baum folgender Formel:

(a + b)/c – 3d · 2a

Lesen Sie danach aus dem Baum die Präfix- und Postfixschreibweise der Formel ab und schreiben Sie diese auf. Beschreiben Sie kurz, wie Sie zu ihrem Ergebnis gekommen sind.

Lösung:



Präfixschreibweise: - / + a b c * * 3 d * 2 a
Postfixschreibweise: a b + c / 3 d * 2 a * * -

Idee Präfix: Schreibe jeden Knoten beim ersten Besuch auf.
Idee Postfix: Schreibe jeden Knoten erst beim letzten Versuch auf.

Hinweis für Tutoren:
Kantorowitsch-Baum (1T), Präfix- und Postfixschreibweisen (je 1T), Erklärungen (je 1T)

Aufgabe 4: Halbgruppen und Monoide (12T)

Überprüfen Sie, ob es sich bei folgenden Strukturen um Halbgruppen oder Monoide oder nichts von beidem handelt. Überprüfen Sie weiterhin, falls es sich um eine Halbgruppe oder einen Monoid handelt, ob die Struktur auch abelsch ist.
(*Hinweis:* Sie dürfen zum Beweisen auf schon bekannte Eigenschaften aus der Vorlesung zurückgreifen. Zum Widerlegen genügt ein begründetes Gegenbeispiel)

4.1 Vektoraddition (2T)

Verknüfung „+“: (a,b) + (c,d) = (a + c, b + d) alle Symbole $\in \mathbb{N}_0$

Lösung:
Die Struktur ist ein abelscher Monoid.

In beiden Stellen des Vektors findet unabhängig voneinander eine normale Addition auf den natürlichen Zahlen statt. Von dieser ist bekannt, dass sie abgeschlossen, assoziativ als auch kommutativ ist und ein Einselement, die Null, besitzt.

4.2 Linksidentität (3T)

Verknüpfung „#“: a # b = a alle Symbole $\in \mathbb{N}_0$

Lösung:
Die Struktur ist eine Halbgruppe.

Abgeschlossen: klar
Assoziativ: a # (b # c) = a # b = a = a # c = (a # b) # c
Kommutativ: nicht gegeben, z.B. a # b = a ≠ b # a = b
Eins: nicht gegeben, z.B. a # e = a ⇒ jedes e $\in \mathbb{N}_0$ ist Rechtseins,
aber e # a = e ⇒ kein e $\in \mathbb{N}_0$ ist Linkseins

4.3 Fließkommazahlen (3T)

Addition auf den Fließkommazahlen doppelter Genauigkeit nach IEEE-754

Lösung:
Die Struktur ist weder eine Halbgruppe, noch ein Monoid.

Abgeschlossen: ja, da auch NaN legitim ist
Assoziativ: nicht gegeben, z.B. $(2^{150} + -2^{150}) + 1 = 1 \neq 0 = 2^{150} + (-2^{150} + 1)$
(hier kann dann abgebrochen werden, da ohne die Assoziativität weder Halbgruppe noch Monoid gegeben ist)

4.4 Matrizenmultiplikation (4T)

Verknüfung „·“: $\begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot \begin{pmatrix} e & f \\ g & h \end{pmatrix} = \begin{pmatrix} ae+bg & af+bh \\ ce+dg & cf+dh \end{pmatrix}$ alle Symbole $\in \mathbb{N}_0$

Falls die Struktur nicht abelsch ist, existiert eine abelsche Unterhalbgruppe / ein abelscher Untermonoid ? Wenn ja, geben Sie dies an. (1T)

Lösung:
Die Struktur ist weder ein Monoid.

Abgeschlossen: klar
Assoziativ: erfüllt

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot (\begin{pmatrix} e & f \\ g & h \end{pmatrix} \cdot \begin{pmatrix} i & j \\ k & l \end{pmatrix}) = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot \begin{pmatrix} ei+fk & ej+fl \\ gi+hk & gj+hl \end{pmatrix} = \begin{pmatrix} aei+afk+bgi+bhk & aej+afl+bgj+bhl \\ cei+cfk+dgi+dhk & cej+cfl+dgj+dhl \end{pmatrix}$$

$$(\begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot \begin{pmatrix} e & f \\ g & h \end{pmatrix}) \cdot \begin{pmatrix} i & j \\ k & l \end{pmatrix} = \begin{pmatrix} ae+bg & af+bh \\ ce+dg & cf+dh \end{pmatrix} \cdot \begin{pmatrix} i & j \\ k & l \end{pmatrix} = \begin{pmatrix} aei+afk+bgi+bhk & aej+afl+bgj+bhl \\ cei+cfk+dgi+dhk & cej+cfl+dgj+dhl \end{pmatrix}$$

Kommutativ: nicht erfüllt

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot \begin{pmatrix} e & f \\ g & h \end{pmatrix} = \begin{pmatrix} ae+bg & af+bh \\ ce+dg & cf+dh \end{pmatrix} \neq \begin{pmatrix} ae+cf & be+df \\ ag+cg & bg+dh \end{pmatrix} = \begin{pmatrix} e & f \\ g & h \end{pmatrix} \cdot \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

Eins: erfüllt

Ann.: $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ ist das Einselement.

Test: $\begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} a1+b0 & a0+b1 \\ c1+d0 & c0+d1 \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{pmatrix} 1a+0b & 0a+1b \\ 1c+0d & 0c+1d \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

Der Untermonoid auf der Untermenge $U = \left\{ \begin{pmatrix} a & 0 \\ 0 & b \end{pmatrix} | a,b \in N_0 \right\}$ ist abelsch.

Beweis: $\begin{pmatrix} a & 0 \\ 0 & b \end{pmatrix} \cdot \begin{pmatrix} c & 0 \\ 0 & d \end{pmatrix} = \begin{pmatrix} ac & 0 \\ 0 & bd \end{pmatrix} = \begin{pmatrix} ca & 0 \\ 0 & db \end{pmatrix} = \begin{pmatrix} c & 0 \\ 0 & d \end{pmatrix} \cdot \begin{pmatrix} a & 0 \\ 0 & b \end{pmatrix}$