

Einführung zur Aufgabengruppe 3

- **Rekursionen**
- **Backus-Notation und Syntaxanalyse**

Beispiel Türme von Hanoi

Türme von Hanoi des Monsieur Claus, Anagramm des Mathematikers Lucas (vor 100 Jahren)

Legende: Zu Benares in Indien gibt es einen Tempel, in dem seit vielen Jahrhunderten Priester bemüht sind, einen Turm so umzulegen, wie Brahma es ihnen selbst geboten hat. Der Turm besteht aus 64 der Größe nach geordneten und auf einer Stange aufgesteckten dünnen goldenen Plättchen. Die Aufgabe lautet, den wenige Zentimeter hohen Turm von der ersten Stange auf die letzte Stange der drei zur Verfügung stehenden Stangen zu schaffen. Dabei darf jeweils nur eine Scheibe transportiert werden und niemals eine größere Scheibe auf eine kleinere gelegt werden. Sobald die Mönche dieses Ritual hinter sich gebracht haben, wird der Tempel zu Staub zerfallen und die Welt mit einem Donnerschlag untergehen! Wann wird wohl mit der indischen Götterdämmerung zu rechnen sein?

Trotz der indischen Geschichte ist das Puzzle als „Türme von Hanoi“ bekannt geworden.

void Hanoi(int, int, int);

```
/* **** */
/* Hanoi: rekursive Funktion */
/* Parameter: k .. Anzahl der Scheiben */
/*          a .. Startstange */
/*          b .. Zielstange */
/*  1. (k-1) Scheiben von Stange a nach Stange c */
/*  2. k. Scheibe von Stange a nach Stange b */
/*  3. (k-1) Scheiben von Stange c nach Stange b */
/* **** */

void Hanoi( int k, int a, int b)
{ if( k > 0)
  { Hanoi( k - 1, a, 6 - a - b); /* 1. */
    printf("%3i:%i => %i", k, a, b); /* 2. */
    Hanoi( k - 1, 6 - a - b, b); /* 3. */
  }

  return;
}
```

Vermutung: Bei n Scheiben sind $f(n) = 2^n - 1$ Bewegungen erforderlich.

*Man benötigt bei n Scheiben mindestens $2^n - 1$ Züge, um das Problem zu lösen.
Das sind bei 64 Scheiben 18'446'744'073'709'551'615 Aktionen.*

Die Priester werden wohl noch Milliarden von Jahren zu tun haben.

Backus-Notation und Syntaxanalyse

Computer verwenden zur Verarbeitung von Daten und Informationen künstliche, formale Sprachen:

- Maschinensprachen
- Assemblersprachen
- Programmiersprachen
- Datenbanksprachen
- ...

Vorteile gegenüber natürlichen Sprachen:

- exakte Definition der Syntax
- keine Kontextabhängigkeit der Bedeutung
(Die Katze, die die Maus frisst. Wer frisst wen?)

Aufbau formaler Sprachen:

- Alphabet Σ : Terminale der Sprache
- Sprache $L \subseteq \Sigma^*$: Teilmenge aller endlichen Wörter, die aus dem Alphabet durch Hintereinanderschreiben seiner Elemente entstehen können

Beschreibung formaler Sprachen:

- Syntax legt Teilmengenbeziehung fest (welche Wörter aus Σ^* zu L gehören)
- Semantik legt deren Bedeutung fest

Syntaxbeschreibung:

- endliche Sprachen: Aufzählung der Sprachelemente (Maschinensprachen)
- unendliche Sprachen: endliche Beschreibung

Backus-Notation als Beschreibungssprache

Im Zusammenhang mit der Entstehung der Programmiersprache ALGOL 60 wurde Ende der fünfziger Jahre des letzten Jh. die Backus-Form (BF) als Beschreibungssprache für kontextfreie Sprachen entwickelt, welche später zur Erweiterten Backus-Naur-Form (EBNF) ergänzt wurde.

x Terminal x (Alphabetzeichen)

$\langle x \rangle$ Nichtterminal (Variable der Sprachbeschreibungssprache)

$\langle x \rangle ::= X$ dem Nichtterminal $\langle x \rangle$ wird die Sprachkonstruktion X zugewiesen

Metasymbole in der Sprachkonstruktion X der Erweiterten Backus-Naur-Form (EBNF):

$X | Y$ Auswahl der Sprachkonstruktion X oder Y

$[X]$ optionales Auftreten der Sprachkonstruktion X

$\{ X \}$ wenn die Sprachkonstruktion X auftritt, so auch mehrfach möglich

Syntaxanalyse von vollständig geklammerten arithmetischen Ausdrücken

Erweiterte Backus-Naur-Form:

<K>	::=	0		1		2		3		4		5		6		7		8		9		<K><K>
<Op>	::=	+		-		*		/		^												
<Fkt>	::=	-		sin		cos		tan		lg		ln										
<Aus>	::=	x		<K>		<FKT><AUS>		(<AUS><OP><AUS>)												

1. Ausführung des Programms Syntax mit der Eingabe von $((\sin x / \lg(2^x)) - (x^3))$

Eingabe eines Terms aus <AUS>: $((\sin x / \lg(2^x)) - (x^3))$

Syntaxanalyse $((\sin x / \lg(2^x)) - (x^3))$

Untersuche Zeichenfolge $((\sin x / \lg(2^x)) - (x^3))$

Untersuche Zeichenfolge $(\sin x / \lg(2^x)) - (x^3)$

Untersuche Zeichenfolge $\sin x / \lg(2^x) - (x^3)$

Untersuche Zeichenfolge $x / \lg(2^x) - (x^3)$

Untersuche Zeichenfolge $\lg(2^x) - (x^3)$

Untersuche Zeichenfolge $(2^x) - (x^3)$

Untersuche Zeichenfolge $2^x - (x^3)$

Untersuche Zeichenfolge $x - (x^3)$

Untersuche Zeichenfolge (x^3)

Untersuche Zeichenfolge x^3

Untersuche Zeichenfolge 3

Syntax korrekt!

2. Ausführung des Programms Syntax mit der Eingabe von $((\sin x / \lg(2x)) - (x^3))$

Eingabe eines Terms aus <AUS>: $((\sin x / \lg(2x)) - (x^3))$

Syntaxanalyse $((\sin x / \lg(2x)) - (x^3))$

Untersuche Zeichenfolge $((\sin x / \lg(2x)) - (x^3))$

Untersuche Zeichenfolge $(\sin x / \lg(2x)) - (x^3)$

Untersuche Zeichenfolge $\sin x / \lg(2x) - (x^3)$

Untersuche Zeichenfolge $x / \lg(2x) - (x^3)$

Untersuche Zeichenfolge $\lg(2x) - (x^3)$

Untersuche Zeichenfolge $(2x) - (x^3)$

Untersuche Zeichenfolge $2x - (x^3)$

Syntaxfehler