Einführung in die Theoretische Informatik
Tutorium XI

Michael R. Jung

15. – 20. 01. 2016

$\mathrm{num}(x), \mathrm{str}(n)$

Sei $\Sigma = \{a_0, \dots, a_{m-1}\}$, also $|\Sigma| = m$ und sei
$x = a_{i_1} a_{i_2} \dots a_{i_n} \in \Sigma^n \subseteq \Sigma^*$.

- Nun ist $\mathrm{num}_\Sigma(x) := \sum_{j=0}^{n-1} m^j + \sum_{j=1}^{n} i_j m^{n-j} = \frac{m^n-1}{m-1} + (i_1 \dots i_n)_m$, dabei gibt $\frac{m^n-1}{m-1}$ den Offset für Wörter der Länge n an und man addiert $k = (i_1 \dots i_n)_m$, d.h. x ist einfach das lexikographisch $(k+1)$-te Wort der Länge n.
- Da $\mathrm{num}_\Sigma : \Sigma^* \to \mathbb{N}$ bijektiv ist, können wir $\mathrm{str}_\Sigma : \mathbb{N} \to \Sigma^*$ definieren als $\mathrm{str}_\Sigma(n) := \mathrm{num}_\Sigma^{-1}(n)$.

Wenn Σ aus dem Zusammenhang klar ist, schreiben wir statt num_Σ oder str_Σ nur num bzw. str.

Beispiele für $\Sigma = \{0, 1\}$:

x	0	10	00	ε	00101
$\text{num}(x)$	1	5	3	0	36

n	2	4	12	32	115
$\text{str}(n)$	1	01	101	00001	110100

Sei $\Sigma = \{0, 1, \dots, 9\}$.

Aufgabe 1

Geben Sie $\mathrm{str}(3), \mathrm{str}(17), \mathrm{str}(100), \mathrm{str}(120)$ sowie $\mathrm{num}(0), \mathrm{num}(1), \mathrm{num}(10), \mathrm{num}(100), \mathrm{num}(1000)$ an.

Lösung:

n	3	17	100	120
$\mathrm{str}(n)$	2	06	89	009

x	0	1	10	100	1000
$\mathrm{num}(x)$	1	2	21	211	2111

Aufgabe 2

Geben Sie jeweils ein LOOP-, WHILE- und GOTO-Programm für die Funktion

$$f : \mathbb{N} \times \mathbb{N} \to \mathbb{N},\ f(a, b) := a^b$$

an.

```
r_0=r_0+1;
LOOP r_2 DO
  LOOP r_0 DO
    LOOP r_1 DO
      r_3=r_3+1
    END
  END;
  r_0=r_3+0;
  r_3=r_4+0
END
```

Besser: Teilprobleme lösen, auf die man zurückgreifen kann.

```
+(a,b):
r_0=r_1+0;
LOOP r_2 DO
  r_0=r_0+1
END

*(a,b):
LOOP r_1 DO
  r_0=r0+r_2
END
```

Dann kann man ein LOOP-Programm für $f(a, b) = a^b$ folgendermaßen schreiben:

```
r_0=r_0+1;
LOOP r_2 DO
  r_0=r_0*r_1
END
```

Frage: Wie kann ich eine While-Schleife der Form

```
WHILE r != 0 DO
  P
END
```

in eine Schleife der Form

```
WHILE r' = 0 DO
  P
END
```

umwandeln?

Lösung:

```
IF r=0 THEN r'=1 ELSE r'=0;
WHILE r' = 0 DO
  P;
IF r=0 THEN r'=1 ELSE r'=0
END
```

```
+(a,b):
r_0=r_1+0;
WHILE r_2 != 0 DO
  r_0=r_0+1;
  r_2=r_2-1
END

*(a,b):
WHILE r_2 != 0 DO
  r_0=r0+r_1;
  r_2=r_2-1
END
```

Nun kann man ein WHILE-Programm für $f(a, b) = a^b$ folgendermaßen schreiben:

```
r_0=r_0+1;
WHILE r_2 != 0 DO
  r_0=r_0*r_1;
  r_2=r_2-1
END
```

```
+(a,b):
1: r_0=r_1+0
2: IF r_2 = 0 THEN GOTO 6
3: r_0=r_0+1
4: r_2=r_2-1
5: GOTO 2
6: HALT

*(a,b):
1: IF r_2 = 0 THEN GOTO 5
2: r_0=r_0+r_1;
3: r_2=r_2-1
4: GOTO 1
5: HALT
```

Nun kann man ein GOTO-Programm für $f(a, b) = a^b$ folgendermaßen schreiben:

```
1: r_0=r_0+1
2: IF r_2 = 0 THEN GOTO 6
3: r_0=r_0*r_1
4: r_2=r_2-1
5: GOTO 2
6: HALT
```