Beispiel einer nicht LOOP-berechenbaren Funktion

$$\begin{aligned} F(n) &= 1 + \max\{f_P^1(n) \mid P \text{ ist LOOP-Programm mit höchstens} \\ &\qquad n \text{ syntaktischen Komponenten}\} \\ &= 1 + \max\{f_P^1(n) \mid P \text{ ist LOOP-Programm mit höchstens} \\ &\qquad n \text{ syntaktischen Komponenten und} \\ &\qquad \text{Variablen aus } \{x_0, x_1, \ldots, x_n\}\} \end{aligned}$$

- Die Beschränkung auf Variablen $\{x_0, x_1, \ldots, x_n\}$ ist möglich, da ein Programm P mit n syntaktischen Komponenten weniger als n Variablen enthält. Kommt in P eine Variable x_j mit $j > n$ vor, so gibt es ein i mit $2 \leq i \leq n$, so dass x_i nicht im Programm auftritt. Jedes Vorkommen von x_j kann durch x_i ersetzt werden, ohne die berechnete Funktion f_P^1 zu verändern.

- F ist total und im intuitiven Sinne berechenbar.
  (Man berechnet einfach für die endlich vielen Programmme die Ausgabe für die Eingabe $x_1 = n$, bildet das Maximum der endlichen Menge der Ausgaben und addiert 1.)

Satz: F ist nicht LOOP-berechenbar.

Beweis. Angenommen, F wäre LOOP-berechenbar.
Dann gäbe es ein LOOP-Programm P^* mit $F = f^1_{P^*}$.
Sei k die Anzahl der syntaktischen Komponenten in P^*. Es folgt:
$F(k) = f^1_{P^*}(k)$ (wegen $F = f^1_{P^*}$) und
$F(k) \geq 1 + f^1_{P^*}(k)$ (nach Definition von F)

Widerspruch, d.h. Annahme ist falsch

Die Ackermannfunktion

Die Ackermannfunktion $A : \mathbb{N}^2 \to \mathbb{N}$ ist wie folgt rekursiv definiert.

$$
\begin{aligned}
A(0, m) &:= m + 1 \\
A(n + 1, 0) &:= A(n, 1) \\
A(n + 1, m + 1) &:= A(n, A(n + 1, m))
\end{aligned}
$$

- Offensichtlich ist A total und intuitiv berechenbar.
- Man kann zeigen, dass für jede LOOP-berechenbare Funktion $f : \mathbb{N} \to \mathbb{N}$ gilt:
  Es gibt ein $k \in \mathbb{N}$, so dass $f(n) < A(n, n)$ für alle $n \geq k$.
- Damit ist $A_1 : \mathbb{N} \to \mathbb{N}$ mit $A_1(n) = A(n, n)$ nicht LOOP-berechenbar.

Weitere Informationen im Buch von Schöning.