Berechnungsmodelle

Welche Funktionen aus $\mathbb{N}_0^k \to \mathbb{N}_0$ lassen sich mit welchen Algorithmen / Maschinen berechnen?

Turingmaschinen: Bandinhalte vor und nach der Berechnung

Stellen Funktionen aus $\Sigma^* \to \Sigma^*$ dar. Kann man aber mit $\Sigma = \{0, 1, \#\}$ leicht auf $f: \mathbb{N}_0^k \to \mathbb{N}_0$ übertragen.

Bsp.: TM M für $f(x,y,z) = x \cdot (y+z)$ (*)

$x = 5$ $\quad$ $y = 3$ $\quad$ $z = 9$

...	□	□	1	0	1	#	1	1	#	1	0	0	1	□	□	...

$\uparrow$ q_0

Nach endlich vielen Schritten

$f(x,y,z) = 60$

...	□	□	1	1	1	1	0	0	□	□	□	□	□	□	□	...

$\uparrow$ $q_f \in F$

(*) Viel Spaß beim Implementieren! ;-)

LOOP - Programme :

$$
\begin{aligned}
P \;\longrightarrow\;& X := X + C \\
\mid\;& X := X - C \qquad (\leftarrow \text{Null, falls } C > X) \\
\mid\;& P ; P \\
\mid\;& \underbrace{\text{LOOP } X \text{ DO } P \text{ END}}
\end{aligned}
$$

Anzahl an Ausführungen von P ist fest. Egal ob P den Wert von X verändert!

$X \rightarrow x_0 \mid x_1 \mid x_2 \mid \dots$

$C \rightarrow 0 \mid 1 \mid 2 \mid \dots$

P berechnet $f : \mathbb{N}_0^k \rightarrow \mathbb{N}_0$, $(n_1, \dots, n_k) \mapsto f(n_1, \dots, n_k)$, falls

$x_0 = 0$, $x_1 = n_1$, $x_2 = n_2$, ... , $x_k = n_k$, $x_{k+1} = 0$, $x_{k+2} = 0$, ...

Nach Ausführung von P

$x_0 = f(n_1, \dots, n_k)$, Rest: egal!

<u>Bsp.</u>: $f(x) = x^2$

```
P = LOOP x1 DO
        LOOP x1 DO
            x0 := x0 + 1
        END
    END
```

„$x_0 := x_0 + x_1$"

x_0: output , x_1: input , Rest: unbenutzt

Bsp. : $f(x,y) = \max\{x,y\}$

$P = \; x_0 := x_2 + 0 ;$

LOOP x_2 DO

$x_1 := x_1 - 1$

END;

LOOP x_1 DO

$x_0 := x_0 + 1$

END

x_0 : output , x_1 und x_2 : input , Rest: unbenutzt

WHILE - Programme :

$$
\left.\begin{aligned}
P &\to X := X + C \\
&\mid X := X - C \\
&\mid P ; P \\
&\mid \text{WHILE } X \neq 0 \text{ DO } P \text{ END} \\
\\
X &\to x_0 \mid x_1 \mid x_2 \mid \dots \\
C &\to 0 \mid 1 \mid 2 \mid \dots
\end{aligned}\right\}
$$

analog !

GOTO - Programme :

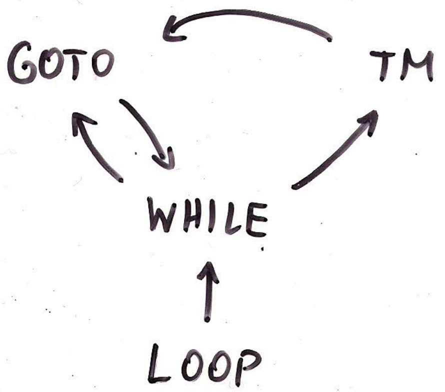
Ähnlich wie LOOP und WHILE, aber mit markierten Anweisungen :

$$P = M_1 : A \; ; \; M_2 : A \; ; \; M_3 : A \; ; \; M_4 : A \; ; \; \ldots$$

$$\begin{aligned} A \;\to\;& X := X + C \\ \mid\;& X := X - C \\ \mid\;& \text{GOTO } M \\ \mid\;& \text{IF } X = C \text{ GOTO } M \\ \mid\;& \text{HALT} \end{aligned}$$

$$\begin{aligned} M &\to M_1 \mid M_2 \mid M_3 \mid \ldots \\ X &\to x_0 \mid x_1 \mid x_2 \mid \ldots \\ C &\to 0 \mid 1 \mid 2 \mid \ldots \end{aligned}$$

Übersetzungen :



$A \to B$ heißt : jedes A-Programm lässt sich in ein äquivalentes B-Programm umwandeln.

D.h. was A-Programme können können B-Programme auch.