Die umgekehrte Richtung ist der berühmte Gödel'sche Vollständigkeitssatz der Prädikatenlogik.

Satz 2.8 (ohne Beweis)

Wenn $\Delta \models \phi$, dann $\Delta \vdash \phi$.

3. Theoretische Berechenbarkeit

3.1 Der Begriff Algorithmus

Intuitiv berechenbar ist ein Funktion f, falls es einen Algorithmus gibt, der für ein beliebiges x aus dem Definitionsbereich von f den Funktionswert $f(x)$ berechnet.

Algorithmus $\hat{=}$ Rechenverfahren, für das gilt:

1. Rechenvorschrift besteht aus endlichem Text.
2. Ablauf einer Berechnung geschieht schrittweise als Folge von elementaren Rechenschritten.
3. In jedem Schritt einer Rechnung ist eindeutig bestimmt, welcher Rechenschritt als nächster durchgeführt wird.
4. Der nächste Rechenschritt hängt nur von der Eingabe und den bisher berechneten Zwischenergebnissen ab.

Satz 3.1

Es existiert eine nichtberechenbare Funktion $f: \mathbb{N} \to \{0,1\}$.

Beweis:

Sei Σ ein endlicher Zeichenvorrat, so dass jeder Algorithmus als Element von Σ^* beschrieben werden kann.

$\Rightarrow$ $\exists$ höchstens abzählbar viele berechenbare Funktionen.

$\Rightarrow$ $\exists$ injektive Abbildung

$$s: \{ f \mid f: \mathbb{N} \to \{0,1\} \text{ berechenbar} \} \longrightarrow \mathbb{N}$$

Sei f_i die i-te berechenbare Funktion.
D.h., $s(f_i) = i$.

Ziel:

Konstruktion einer Funktion $g: \mathbb{N} \to \{0,1\}$, die zu allen Funktionen f_i, $i \in \mathbb{N}$ verschieden ist.

Definiere

$$g(i) = \begin{cases} 1 & \text{falls } f_i(i) = 0 \\ 0 & \text{falls } f_i(i) = 1 \end{cases}$$

Annahme: $g = f_j$ für ein $j \in \mathbb{N}$.

~~Dann~~ Es gilt

$$g(j) \neq f_j(j)$$

Dies ist ein Widerspruch zur Annahme

$\Rightarrow$ Annahme ist falsch

$\Rightarrow$ g ist nicht berechenbar. ∎

Übung: Σ^* ist aufzählbar.

- Diagonalverfahren oder Diagonalisierung

Ziel: exakte Charakterisierung der Klasse der intuitiv berechenbaren Funktionen

3.2 Die μ-rekursiven Funktionen

Idee

1. Definiere einige einfache Grundfunktionen, die im intuitiven Sinn berechenbar sind.
2. Definiere einfache Rechenregeln, mittels denen man aus berechenbaren Fkten neue gewinnt.

Die Klasse der primitiv rekursiven Funktionen ist die kleinste Klasse $\mathbb{P}$ von Fkten, die

1. $\forall r, s \in \mathbb{N}_0$ die konstante Funktion

$$c_s^r : \mathbb{N}_0^r \to \mathbb{N}_0 \quad \text{mit} \quad c_s^r(x) = s$$

2. die Nachfolgerfunktion $N : \mathbb{N}_0 \to \mathbb{N}_0$ mit

$$N(x) = x+1 \quad \text{und}$$

3. $\forall r \in \mathbb{N} \ \forall 1 \leq i \leq r$ die Projektion

$$p_i^r : \mathbb{N}_0^r \to \mathbb{N}_0 \text{ mit } p_i^r(x_1, \dots, x_r) = x_i$$

enthält und unter den nachfolgend definierten Operationen abgeschlossen ist:

- Eine Fkt. $h: \mathbb{N}_0^m \to \mathbb{N}_0$ entsteht durch Substitution der Fkt.en $g_1, g_2, \dots, g_r : \mathbb{N}_0^m \to \mathbb{N}_0$ in $f: \mathbb{N}_0^r \to \mathbb{N}_0$, falls
$$h(x) = f(g_1(x), g_2(x), \dots, g_r(x)).$$
- Eine Fkt. $h: \mathbb{N}_0^{r+1} \to \mathbb{N}_0$ entsteht durch primitive Rekursion aus $g: \mathbb{N}_0^r \to \mathbb{N}_0$ und $f: \mathbb{N}_0^{r+2} \to \mathbb{N}_0$, falls $\forall x \in \mathbb{N}_0^r, \forall n \in \mathbb{N}_0$
$$h(0,x) = g(x) \quad \text{und}$$
$$h(n+1,x) = f(n, h(n,x), x).$$

muß nicht von allen drei Komponenten abhängen.

Lemma 3.1

Eine Fkt. f ist genau dann primitiv rekursiv, wenn es eine endliche Folge $f_1, f_2, \dots, f_s$ von Fkten gibt, für die gilt:

1. $f = f_s$.
2. $\forall\, i \leq s$ gilt:
   a) f_i ist eine der oben definierten Grundfkten oder
   b) f_i entsteht direkt aus einigen f_j, $j < i$ durch Substitution oder primitiver Rekursion.

$f_1, f_2, \dots, f_s$ heißt Ableitung der Funktion f.

08.12.

Bsp. 3.1

- $f: \mathbb{N}_0^2 \to \mathbb{N}_0 \qquad f(x,y) = x+y$

  primitiv rekursiv, da f_1, f_2, f_3 mit

  $f_1(x) = p_1^1(x)$
  $f_2(x) = N(x)$
  $f_3(0,y) = f_1(y), \quad f_3(x+1,y) = f_2(f_3(x,y))$

  Ableitung von f.

- Vorgängerfunktion $\quad V: \mathbb{N}_0 \to \mathbb{N}_0$

$$V(x) = \begin{cases} 0 & \text{falls } x = 0 \\ x-1 & \text{sonst} \end{cases}$$

  primitiv rekursiv, da f_1, f_2, f_3 mit

  $f_1 = C_0^1$
  $f_2(x) = p_1^1(x)$
  $f_3(0) = f_1 \qquad f_3(x+1) = f_2(x)$

  Ableitung von V ist.

- modifizierte Differenz $\quad x \dot{-} y$, wobei

$$x \dot{-} y = \begin{cases} x-y & \text{falls } x-y > 0 \\ 0 & \text{sonst} \end{cases}$$

primitiv rekursiv, oder

$$x \dot{-} 0 = p_1^1(x)$$
$$x \dot{-} (y+1) = V(x \dot{-} y)$$

Eigenschaften können mittels Prädikate beschrieben werden. Ein r-stelliges Prädikat P über $\mathbb{N}_0$ ist eine Teilmenge von $\mathbb{N}_0^r$.

Schreibweise:

$P(x_1, x_2, \ldots, x_r)$ anstatt $(x_1, x_2, \ldots, x_r) \in P$

charakteristische Funktion eines r-stelligen Prädikates P:

$c_P : \mathbb{N}_0^r \to \mathbb{N}_0$, wobei

$$c_P(x_1, x_2, \ldots, x_r) = \begin{cases} 1 & \text{falls } P(x_1, x_2, \ldots, x_r) \\ 0 & \text{sonst.} \end{cases}$$

P heißt primitiv rekursiv, falls c_P primitiv rekursiv ist.

P, Q r-stellige Prädikate

$P \wedge Q \triangleq$ Menge $P \cap Q$
$P \vee Q \triangleq$ Menge $P \cup Q$
$\neg P \triangleq$ Menge $N_0^r \setminus P$

Lemma 3.2

Seien P und Q primitiv rekursive r-stellige Prädikate. Dann sind auch die Prädikate $P \wedge Q$, $P \vee Q$ und $\neg P$ primitiv rekursiv.

Beweis:

Es gilt:

$$c_{P \wedge Q}(x_1, \dots, x_r) = c_P(x_1, \dots, x_r) \cdot c_Q(x_1, \dots, x_r)$$

$$c_{P \vee Q}(x_1, \dots, x_r) = sg\big(c_P(x_1, \dots, x_r) + c_Q(x_1, \dots, x_r)\big)$$

$$c_{\neg P}(x_1, \dots, x_r) = 1 \dot{-} c_P(x_1, \dots, x_r)$$

$\blacksquare$

Satz 3.2

Seien $P_1, P_2, \dots, P_k$ paarweise disjunkte r-stellige primitiv rekursive Prädikate und seien $f_1, \dots, f_k : \mathbb{N}_0^r \to \mathbb{N}_0$ primitiv rekursive Funktionen. Dann ist die Funktion $g: \mathbb{N}_0^r \to \mathbb{N}_0$ mit

$$g(x_1, \dots, x_r) = \begin{cases} f_1(x_1, \dots, x_r) & \text{falls } P_1(x_1, \dots, x_r) \\ f_2(x_1, \dots, x_r) & \text{falls } P_2(x_1, \dots, x_r) \\ \vdots & \\ f_k(x_1, \dots, x_r) & \text{falls } P_k(x_1, \dots, x_r) \\ 0 & \text{sonst} \end{cases}$$

primitiv rekursiv.

Beweis:

Es gilt

$$g(x_1,\dots,x_r) = \sum_{i=1}^{\ell} c_{p_i}(x_1,\dots,x_r)\cdot f_i(x_1,\dots,x_r).$$

$\blacksquare$

Abbildungen, die Fkten in Fkten abbilden, heißen Operatoren.

$f: N_0^{r+1} \to N_0$.

$\mu_b f: N_0^{r+1} \to \mathbb{N}_0$ definiert durch

$$\mu_b f(n, x_1,\dots,x_r) = \begin{cases} \min\{m \mid m \le n \wedge f(m,x_1,\dots,x_r) \neq 0\} & \text{falls solches } m \in \mathbb{N}_0 \text{ existiert} \\ 0 & \text{sonst} \end{cases}$$

beschränkter μ-Operator

Satz 3.3

Sei $f: N_0^{r+1} \to \mathbb{N}_0$ primitiv rekursiv. Dann ist auch $\mu_b f: \mathbb{N}_0^{r+1} \to \mathbb{N}_0$ primitiv rekursiv.

Beweis:

Es gilt:

$$\mu_b f(0,x) = 0 \quad \forall x \in \mathbb{N}_0^r \quad \text{und}$$

$$\mu_b f(n+1,x) = \begin{cases} \mu_b f(n,x) & \text{falls } \mu_b f(n,x) \neq 0 \\ n+1 & \text{falls } \mu_b f(n,x) = 0 \\ & \wedge f(0,x) \neq 0 \wedge f(n+1,x) = 0 \\ 0 & \text{sonst} \end{cases}$$

Betrachte folgende Funktion

$g: \mathbb{N}_0^{r+2} \longrightarrow \mathbb{N}_0$ mit

$$g(n,x,t) = \begin{cases} t & \text{falls ein } z \text{ mit } 0 \leq z \leq n \\ & \text{und } f(z,x) = 0 \text{ existiert} \\ n+1 & \text{falls kein solches } z \text{ existiert} \\ & \text{und } f(n+1,x) = 0 \\ 0 & \text{sonst} \end{cases}$$

Satz 3.2 $\Rightarrow$ g ist primitiv rekursiv.

Nun können wir $\mu_b f$ wie folgt schreiben:

$$\mu_b f(0,x) = 0 \quad \forall x \in \mathbb{N}_0^r \quad \text{und}$$

$$\mu_b f(n+1,x) = g(n,x,f(n,x)).$$

$\Rightarrow$

$\mu_b f$ ist primitiv rekursiv.

∎

Satz 3.4

Es gibt eine intuitiv berechenbare Funktion, die nicht primitiv rekursiv ist.

Beweis:

- mittels Diagonalisierung

Ziel:

Überlegung, dass jede Ableitung einer primitiv rekursiven Funktion als String über einem festen Alphabet Σ dargestellt werden kann.

Σ besteht aus:

- c, N, p, f, =, (,), +, 0, 1, x, n,
- hochgestellten und tiefgestellten Ziffern 0, 1, …, 9 zum indizieren von Funktionssymbolen c, p, f und Variablen x,
- ", " und Zwischenraum ␣.

$\leadsto$

Jede Ableitung kann als String über Σ dargestellt werden. Ferner gilt:

1. Für ein gegebenes $w \in \Sigma^+$ kann durch ein mechanisches Rechenverfahren entschieden werden, ob w die Ableitung einer primitiv rekursiven Funktion $f: \mathbb{N}_0 \to \mathbb{N}_0$ kodiert.
2. Man kann alle Strings in Σ^+ aufzählen, indem man alle Strings der Länge eins, dann alle

Strings der Länge zwei u.s.w. geordnet generiert.

3. Für ein gegebenes $i \in \mathbb{N}_0$ kann die i-te Ableitung G_i, die aufgezählt wird, durch folgendes Verfahren effektiv berechnet werden:

   (a) Zähle Σ_1^+ auf und entscheide für jeden aufgezählten String w, ob dieser eine Ableitung einer primitiv rekursiven Funktion kodiert.

   (b) Sobald die i-te Ableitung gefunden ist, halte mit dieser an.

Sei g_i die durch G_i kodierte primitiv rekursive Funktion. Sei $h : \mathbb{N}_0 \to \mathbb{N}_0$ definiert durch

$$h(i) := g_i(i) + 1$$

h besitzt folgende Eigenschaften:

1) h ist im intuitiven Sinn berechenbar.
2) h ist nicht primitiv rekursiv.
(Andernfalls $\exists$ Ableitung H von h und $j \in \mathbb{N}$ mit $H = G_j$. Dann wäre

$$g_j(j) = h(j) = g_j(j) + 1$$

Widerspruch) ∎

Der Beweis verwendet zwei Eigenschaften der primitiv rekursiven Funktionen:

1) Alle "Programme" (Ableitungen) primitiv

rekursiver Funktionen können effektiv generiert und auf einheitliche Art und Weise ausgewertet werden.

2) Da alle primitiv rekursiven Funktionen <u>total</u> sind (d.h., für jedes Element des Definitionsbereiches ist der dazugehörige Funktionswert definiert), konnte die Diagonalisierung durchgeführt werden.

$\rightsquigarrow$

Die Klasse aller totalen berechenbaren Funktionen kann nicht konstruktiv charakterisiert werden.

Falls wir es zulassen, dass für Elemente aus dem Definitionsbereich der Funktion der dazugehörige Funktion nicht definiert ist, dann haben wir eine <u>partielle Funktion</u>.

<u>Ziel</u>:
Konstruktive Charakterisierung der Klasse aller berechenbaren Funktionen (auch partielle).

$\rightsquigarrow$

Definition einer weiteren Operation, die partielle Funktionen erzeugt.

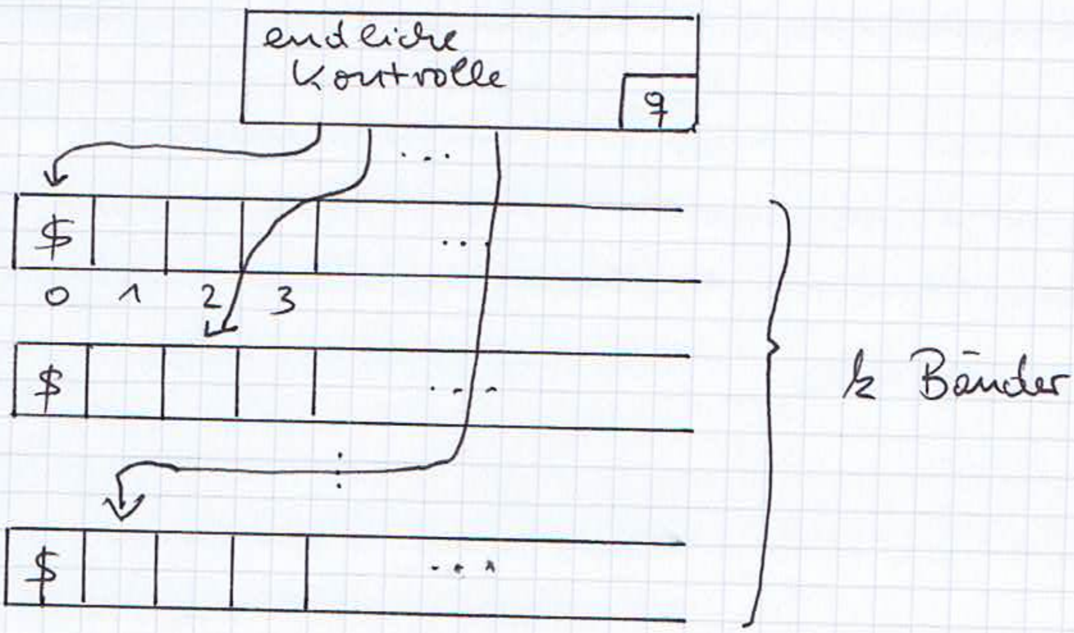
Sei $f: \mathbb{N}_0^{r+1} \longrightarrow \mathbb{N}_0$ eine Funktion. Dann ist $\mu f: \mathbb{N}_0^r \to \mathbb{N}_0$ definiert durch

$$\mu f(x) := \begin{cases} \min\{n \geq 0 \mid f(n,x) = 0 \text{ und } f(m,x) \text{ ist } \forall\, m < n \text{ definiert}\} & \text{falls solches } n \in \mathbb{N}_0 \text{ existiert} \\ \text{nicht definiert} & \text{sonst.} \end{cases}$$

μ heißt <u>μ-Operator</u>. Die Klasse der <u>μ-rekursiven Funktionen</u> ist die kleinste Klasse R von Funktionen, für die gilt:

1. Alle konstanten Funktionen, die Nachfolgerfunktion sowie alle Projektionen sind in R.
2. R ist abgeschlossen unter Substitution, primitiver Rekursion und der Anwendung des μ-Operators.

3.3 <u>Turingmaschinen</u>



Eine (<u>deterministische</u>) <u>k-Band Turingmaschine</u> (DTM) M ist ein 5-Tupel $M = (Q, \Sigma, \delta, q_0, F)$, wobei

1) Q eine endliche Menge von <u>Zuständen</u>,
2) Σ ein endliches <u>Bandalphabet</u> mit $Q \cap \Sigma = \emptyset$ und $\{0, 1, \#, \sqcup, \$\} \subseteq \Sigma$,
3) $\delta: Q \times \Sigma^k \longrightarrow Q \times \Sigma^k \times \{-1, 0, 1\}^k$ die <u>Übergangsfunktion</u>,
4) q_0 der <u>Startzustand</u> und

5) $F := \{ q_e \mid \delta(q_e, a)$ ist für kein $a \in \Sigma^k$ definiert$\}$ die Menge der <u>Endzustände</u>

sind.

- <u>Interpretation von</u>

$$\delta(q, a_1, a_2, \ldots, a_k) = (q', c_1, c_2, \ldots, c_k, s_1, \ldots, s_k)$$

- Zu jedem Zeitpunkt ist DTM eindeutig spezifiziert durch

  - aktuellen Zustand der endlichen Kontrolle,
  - den Bandinhalt der Bänder und
  - Positionen der L/S-Köpfe auf den Bändern.

<u>Konfiguration</u>

- <u>Startkonfiguration</u>

1. q_0 Zustand der endlichen Kontrolle.
2. Eingabe der Länge n steht auf den Bandquadraten $1, 2, \ldots, n$ des ersten Bandes.
3. Alle übrigen Bandquadrate der k Bänder, bis auf Bandquadrat 0, das \$ enthält, sind mit dem Sonderzeichen $\sqcup$ (Blank) beschrieben

4. Alle L/S-Köpfe stehen auf dem ersten Bandquadrat des zugehörigen Bandes.

<u>Randbedingungen:</u>

- L/S-Kopf niemals über das linke Ende des Bandes hinaus bewegen
- Inhalt des Bandquadrates 0 darf niemals überschrieben werden
- Links von L/S-Kopf steht niemals ⊔.

$\Gamma \subseteq \Sigma$ mit der Eigenschaft, dass Γ die kleinste Teilmenge von Σ ist, so dass für alle Eingaben w von M gilt $w \in \Gamma^*$, heißt <u>Eingabealphabet</u> von M.

- <u>Endkonfiguration</u>

  aktuelle Zustand $q \in F$.

- <u>Nachfolgekonfiguration</u>

  $K \vdash_M K'$ K geht mittels einer Anwendung von δ nach K' über

  $\vdash_M^*$ reflexive, transitive Hülle von $\vdash_M$

- Eine endliche Folge $K_0, K_1, \ldots, K_t$ heißt <u>endliche Rechnung von M auf der Eingabe w</u>, falls

1. K_0 Startkonfiguration von M bei der Eingabe w,
2. $K_i \vdash_M K_{i+1}$ für $0 \leq i < t$ und
3. K_t eine Endkonfiguration

sind.

- M <u>hält</u> bei der Eingabe w <u>an</u>, falls eine endliche Rechnung von M auf w gibt.
- TM <u>heißt regulär</u>, falls am Ende jeder endlichen Rechnung alle L/S-Köpfe auf dem Bandquadrat 1 des zugehörigen Bandes stehen.

<u>Ziel</u>:

Spezifikation der durch eine TM M berechneten Funktion.

- $bin(n)$, $n \in \mathbb{N}_0$ Binärdarstellung von n ohne führende Nullen.

D.h.,

- $bin: \mathbb{N}_0 \rightarrow \{0,1\}^+$ injektive Fkt.
- $bin^{-1}: \{0,1\}^+ \rightarrow \mathbb{N}_0$ mit

$$bin^{-1}(w_1 w_2 \ldots w_s) = \sum_{i=1}^{s} w_i 2^{s-i}$$

Umkehrfunktion von bin.

Jede TM M mit Bandalphabet Σ definiert die eventuell partielle Fkt. $f_M: \Sigma^* \rightarrow \Sigma^*$, wobei $f_M(w)$ die Inschrift des ersten Bandes am Ende