9. Universelle Maschinen und universelle Funktionen

9.1 Universelle partiell rekursive Funktionen und universelle Turingmaschinen

In diesem Abschnitt zeigen wir die Existenz universeller Turingmaschinen U. Solch eine Maschine U simuliert (interpretiert) alle Turingmaschinen (genauer: alle geeignet normierten TM zur Berechnung eines gegebenen Typs von Funktionen).

Dies ist die theoretische Grundlage für die Konstruktion von Universalrechnern (Computern).

UNIVERSELLE FUNKTIONEN

DEFINITION. Sei F eine Klasse von (partiellen) Funktionen über $\mathbb{N}$. Eine (partielle) Funktion $\varphi^{(n+1)}$ ist n-*universell* für F, wenn folgendes gilt:

(i) $\varphi \in$ F

(ii) Zu jeder n-stelligen (partiellen) Funktion $\psi^{(n)} \in$ F gibt es eine Zahl $e \in \mathbb{N}$, sodass $\psi(\vec{x}) = \varphi(e, \vec{x})$ für alle $\vec{x} \in \mathbb{N}^n$.

Eine Zahl e wie in (ii) heißt ein *Index von* ψ *bzgl.* φ.

Wir schreiben auch $\varphi_e(\vec{x})$ statt $\varphi(e, \vec{x})$ und nennen φ_e den e-ten *Zweig* von φ.

Eine $n+1$-stellige (partielle) Funktion $\varphi \in$ F ist also genau dann n-universell für F, wenn ihre Zweige gerade alle n-st. (partiellen) Funktionen in F sind.

UNIVERSELLE PARTIELL BERECHENBARE FUNKTIONEN

Eine n-universelle Funktion φ für die Klasse der partiell berechenbaren Funktionen können wir mit Hilfe der Church-Turing-These mit dem folgenden Berechnungsverfahren erhalten. Dabei benutzen wir die im Beweis des Äquivalenzsatzes eingeführte Gödelisierung von (normierten) Turing-Maschinen:

Berechnungsverfahren für φ (*Idee*):

Bei Eingabe $(e, \vec{x}) \in \mathbb{N}^{n+1}$ verfahre wie folgt:

Prüfe, ob e die Gödelnummer $\tilde{M}$ einer TM M ist. Falls nein, gebe $\varphi(e, \vec{x}) = 0$ aus. Falls ja, simuliere M bei Eingabe $\vec{x}$. Liefert diese Simulation das Ergebnis y, so gebe $\varphi(e, \vec{x}) = y$ aus. (Terminiert M nicht, so terminiert die Simulation natürlich ebenfalls nicht und das Ergebnis ist undefiniert: $\varphi(e, \vec{x}) \uparrow$).

Nach der Church-Turing-These (und dem Äquivalenzsatz) folgt hieraus die Existenz universeller partiell rekursiver Funktionen. Wegen der Bedeutung dieses Ergebnisses wollen wir dies jedoch ohne Rückgriff auf die Church-Turing-These zeigen, indem wir obiges Argument formalisieren und so eine universelle Turingmaschine konstruieren.

UNIVERSELLE TURING MASCHINEN

1. SATZ ÜBER DIE EXISTENZ UNIVERSELLER TURINGMASCHINEN. Zu jedem Alphabet Σ und zu jeder Stelligkeit $n \geq 1$ gibt es eine Turingmaschine

$$\tilde{U} = \tilde{U}_{\Sigma,n} = (\Sigma, n+1, \Sigma, \Sigma \cup \{b, 0, 1\}, Z, z_0, \delta)$$

mit folgender Eigenschaft:

Zu jeder partiell Turing-berechenbaren Funktion $\psi : (\Sigma^*)^n \to \Sigma^*$ gibt es ein Wort $w \in \Sigma^*$ mit $\psi = (\varphi_{\tilde{U}})_w$, d.h.,

$$\forall \vec{x} \in (\Sigma^*)^n (\psi(\vec{x}) = \varphi_{\tilde{U}}(w, \vec{x})).$$

Da die Turingberechenbarkeit von zahlentheoretischen Funktionen über die Turingberechenbarkeit der entsprechenden auf den Unärdarstellungen operierenden Wortfunktionen über $\Sigma = \{0\}$ definiert ist, folgt hieraus mit dem Äquivalenzsatz direkt der Satz über die Existenz universeller part. rek. Funktionen.

UNIVERSELLE PARTIELL REKURSIVE FUNKTIONEN

SATZ ÜBER DIE EXISTENZ UNIVERSELLER PART. REK. FUNKTIONEN. Sei $n \geq 1$. Es gibt eine partiell rekursive Funktion

$$\varphi : \mathbb{N}^{n+1} \to \mathbb{N},$$

deren Zweige gerade die n-stelligen partiell rekursiven Funktionen sind. D.h. φ ist n-universell für F(REC).

Wir beweisen den Satz, indem wir die Existenz universeller Turingmaschinen nachweisen. Dabei beschränken wir uns nicht nur auf den benötigten Fall von Turingmaschinen, die zahlentheoretische Funktionen berechnen.

Zum Beweis des 1. Satzes über die Existenz universeller TMs gehen wir wie folgt vor. (Gegeben seien das Ein- und Ausgabealphabet Σ und die Stelligkeit $n \geq 1$.)

- Geeignete Normierung von TMs über dem Alphabet Σ.
- Kodierung normierter TMs M durch Binärwörter $\lceil M \rceil$.
- Konstruktion einer 3-Band-TM U, die bei Eingabe $(\lceil M \rceil, \vec{x})$ die normierte Maschine M bei Eingabe $\vec{x}$ simuliert.
- Konstruktion einer TM BIN, die die Eingabe 0^{m+1} in das $(m+1)$-te Binärwort umwandelt (Binärzähler).

Für Σ mit $\{0, 1\} \subseteq \Sigma$ hat dann $\tilde{U}_{\Sigma,n} = U$ die gewünschten Eigenschaften. Für $\Sigma = \{0\}$ erhält man $\tilde{U}_{\Sigma,n}$ durch Hintereinanderausführung von BIN und U (genauer: einer zu U äquivalenten 1-Band-TM).

Im Folgenden skizzieren wir den Beweis.

NORMIERUNG VON TURINGMASCHINEN

Eine Turingmaschine $M = (\Sigma, n, \Sigma, \Gamma, Z, z_0, \delta)$ zur Berechnung einer n-stelligen partiellen Funktion $\psi : (\Sigma^*)^n \to \Sigma^*$ ist *normiert*, falls

- $\Gamma = \Sigma \cup \{b, 0, 1\}$
- $Z = \{0, 1, \ldots, p\}$ (für $p \geq 1$ geeignet)
- $z_0 = 0$
- 1 ist ausgezeichneter Stoppzustand, d.h. $\delta(z, a) \uparrow \;\Leftrightarrow\; z = 1$.

Die erste Forderung besagt gerade, dass M Bandalphabet-normiert ist. Die anderen Forderungen entsprechen gerade den im Beweis des Äquivalenzsatzes vorgenommenen Normierungen. Aus den an den entsprechenden Stellen gemachten Beobachtungen folgt daher, dass man jeder TM M effektiv eine äquivalente normierte TM M' zuordnen kann, wobei sich Rechenzeit und Platzbedarf von M' wie folgt abschätzen lassen:

$$time_{M'}(\vec{x}) \leq O(time_M(\vec{x})) \;\;\&\;\; space_{M'}(\vec{x}) \leq O(space_M(\vec{x}))$$

EXISTENZ UNIVERSELLER TMs

2. SATZ ÜBER DIE EXISTENZ UNIVERSELLER TURINGMASCHINEN. Sei Σ ein Alphabet und $n \geq 1$. Es gibt eine 3-Band-Turingmaschine $U = U_{\Sigma,n}$, die bei Eingabe

$$(\lceil M \rceil, \vec{x}) \in \{0,1\}^* \times (\Sigma^*)^n,$$

wobei

$$M = (\Sigma, n, \Sigma, \Sigma \cup \{b, 0, 1\}, \{0, \ldots, p\}, 0, \delta)$$

eine normierte TM zur Berechnung einer n-stelligen partiellen Funktion vom Typ $(\Sigma^*)^n \to \Sigma^*$ ist, die Maschine M für Eingabe $\vec{x}$ simuliert, d.h. den Wert

$$\varphi_U(\lceil M \rceil, \vec{x}) = \varphi_M(\vec{x})$$

liefert.

Weiter gilt, dass

$$\text{time}_U(\lceil M \rceil, \vec{x}) \in O(|\lceil M \rceil| \cdot \max(\text{time}_M(\vec{x}), |x_1|, \ldots, |x_n|))$$

$$\text{space}_U(\lceil M \rceil, \vec{x}) \in O(|\lceil M \rceil| + \text{space}_M(\vec{x})).$$

KODIERUNG NORMIERTER TURINGMASCHINEN

Für gegebenes Eingabealphabet Σ und gegebene Stelligkeit $n \geq 1$ unterscheiden sich normierte TMs

$$M = (\Sigma, n, \Sigma, \Sigma \cup \{b, 0, 1\}, \{0, \ldots, p\}, 0, \delta)$$

zur Berechnung von n-stelligen partiellen Funktionen vom Typ $\psi : (\Sigma^*)^n \to \Sigma^*$ nur in der Anzahl der Zustände und der Programmfunktion δ. Da sich die Zustände aus dem Programm ergeben, genügt es daher zur Kodierung von M das Programm δ zu kodieren:

Hierzu kodieren wir zunächst Buchstaben des Bandalphabets, Zustände und Bewegungen unär ($\Gamma = \{a_0 = b, a_1 = 0, a_2 = 1, \ldots, a_m\}$, $m \geq 2$):

$$\lceil a_i \rceil := 0^{i+1} \quad \lceil z \rceil := 0^{z+1} \quad \lceil L \rceil := 0 \quad \lceil S \rceil := 0^2 \quad \lceil R \rceil := 0^3.$$

Jeder Instruktion $I = (z, a_i, a_j, B, z')$ - d.h. Programmzeile $\delta(z, a_i) = (a_j, B, z')$ - ordnen wir dann folgendes Binärwort zu:

$$\lceil I \rceil = \lceil z \rceil 1 \lceil a_i \rceil 1 \lceil a_j \rceil 1 \lceil B \rceil 1 \lceil z' \rceil$$

Durch Auflisten der Programmzeilen $\delta \mathrel{\widehat{=}} \{I_1, \ldots, I_k\}$ erhalten wir dann die Binärkodierung von δ und damit von M:

$$\lceil M \rceil := \lceil \delta \rceil := 11 \lceil I_1 \rceil 11 \lceil I_2 \rceil 11 \ldots \lceil I_k \rceil$$

BEWEISIDEE. Wir verzichten auf eine formale Spezifikation von U und begnügen uns mit einer informellen Beschreibung der Arbeitsweise von U bei Eingabe $(\lceil M \rceil, \vec{x})$.

Die Bänder von U haben folgende Funktion:

Band 1: U verhält sich auf Band 1 wie die simulierte (1-Band-) Maschine M bei Eingabe $\vec{x}$.

Band 2: Hier wird das kodierte Programm $\lceil \delta \rceil$ von M abgespeichert.

Band 3: Hier wird die Kodierung $\lceil z \rceil$ des aktuellen M-Zustandes z gespeichert.

Die Rechnung von U besteht aus drei Phasen, der *Initialisierungsphase*, der *Simulationsphase* und der *Ausgabephase*.

In der *Initialisierungsphase* wird unter Löschen $\lceil M \rceil$ von Band 1 auf Band 2 kopiert und auf Band 3 der kodierte Startzustand $\lceil 0 \rceil$ geschrieben. In der *Ausgabephase* wird nach Abschluss der Simulation die Ausgabe von Band 1 auf (das Ausgabe-)Band 3 kopiert.

In der *Simulationsphase* wird jeder Schritt von M durch eine Folge von Rechenschritten mit Hilfe des folgenden *Simulationszyklus* simuliert.

Simulationszyklus von U zur Simulation eines M-Schrittes:

1. Lese (und merke im Zustand) den Buchstaben a auf dem Arbeitsfeld von Band 1.

2. Durch Mustervergleich finde das erste Vorkommen der Folge $11\lceil z\rceil 1\lceil a\rceil 1$ auf Band 2, wobei $\lceil z\rceil$ die aktuelle Inschrift von Band 3 ist. Dort befindet sich dann die Kodierung der auszuführenden M-Instruktion $\lceil I\rceil = \lceil z\rceil 1\lceil a\rceil 1\lceil a'\rceil 1\lceil B\rceil 1\lceil z'\rceil$. (Findet U das gesuchte Muster nicht, so hat M den Stoppzustand erreicht, und die Simulationsphase wird beendet.)

3. Simuliere diese Instruktion: (a) führe die angegebenen Druck- (a') und Bewegungsbefehle (B) auf Band 1 aus und (b) schreibe die Kodierung $\lceil z'\rceil$ des neuen Zustands auf Band 3.

Die behaupteten Platz- und Zeitschranken lassen sich leicht nachprüfen. Für die Zeit hat man nur zu beachten, dass die Simulation eines M-Schrittes $O(|\lceil M\rceil|)$ Schritte erfordert.

q.e.d.

9.2 Gödelnummerierungen der partiell rekursiven Funktionen und das Normalformtheorem von Kleene

BINÄRZÄHLER

Eine TM BIN, die bei Eingabe 0^{n+1} das $n+1$-te Binärwort w_n auf das 2. Band schreibt, arbeitet wie folgt:

Initialisierung:

BIN liest die erste 0 und schreibt $[\lambda]$ $(= [w_0])$ auf Band 2.

Iterationsphase:

Für jede weitere 0, die BIN auf Band 1 findet, ersetzt es das aktuelle geklammerte Binärwort $[w_i]$ auf Band 2 durch das Wort $[w_{i+1}]$. (Hierzu durchläuft BIN das Wort $[w_i]$ von rechts nach links bis zur ersten 0 und ersetzt dabei alle bis dahin durchlaufenen Einsen durch Nullen und die erste gefundene 0 durch eine 1. Findet BIN keine 0, so ersetzt BIN alle Einsen durch Nullen und fügt links eine weitere 0 an, d.h. ersetzt [durch [0.)

Schlussphase:

Am Ende streicht BIN noch die eckigen Klammern um w_n.

Aus $|z_n| \in O(log(n))$ folgt leicht, dass BIN $O(n \cdot log(n))$-zeitbeschränkt (und $n + O(0)$-platzbeschränkt) ist.

GÖDELNUMMERIERUNGEN VON F(REK)

Die Existenz universeller partiell rekursiver Funktionen lässt sich noch mit Hilfe der folgenden Definition verschärfen.

DEFINITION. Eine n-universelle partielle Funktion φ für F(REK) ist eine *Standardaufzählung* oder *Gödelnummerierung* der n-stelligen partiell rekursiven Funktionen, wenn es zu jeder $(n+1)$-st. partiell rekursiven Funktion $\hat{\varphi}^{(n+1)}$ eine total rekursive Funktion h gibt, so dass

(*) $\hat{\varphi}(e, \vec{x}) = \varphi(h(e), \vec{x})$ für alle $e \in \mathbb{N}$ und $\vec{x} \in \mathbb{N}^n$

gilt.

Die rekursive Funktion h in (*) heißt *Übersetzungsfunktion* von $\hat{\varphi}$ nach φ.

DIE BEDEUTUNG VON GÖDELNUMMERIERUNGEN

Gödelisieren wir statt Turingmaschinen Registermaschinen (oder partiell rekursive Funktionen), um zu diesem Konzept wiederum einen Interpreter zu konstruieren, so liefert das eine alternative n-universelle partiell rekursive Funktion $\hat{\varphi}$.

Da wir (im Rahmen des Beweis des Äquivalenzsatzes) gezeigt haben, dass man zu jeder Registermaschine M effektiv eine äquivalente Turingmaschine M' angeben kann, folgt aus der Effektivität der Gödelisierungen, dass es eine berechenbare - also nach der Church-Turing-These - rekursive Funktion h gibt, die die Gödelnummer von M auf die Gödelnummer von M' abbildet, also $\hat{\varphi}$ nach φ übersetzt.

Da wir zeigen werden, dass die auf der Interpretation von TMs basierende universelle Funktion φ eine Gödelnummerierung ist, gilt diese Beobachtung ganz allgemein. Für *jede* effektive Darstellungsweise von n-st. partiell rekursiven Funktionen (die sich durch geeignete Gödelisierung mit Hilfe einer $n+1$-st. partiell rekursiven Funktion $\hat{\varphi}$ beschreiben lässt) kann man - mit Hilfe der Übersetzungsfunktion von $\hat{\varphi}$ nach φ - zu einer gegebenen Darstellung effektiv eine äquivalente Turingmaschine angeben.

BEWEISIDEE. Für gegebenes $n \geq 1$ sei $\tilde{U} = \tilde{U}_{\Sigma_1,n}$ die universelle Turingmaschine zur Berechnung n-stelliger partieller Funktionen über Σ_1^* und sei $\varphi := \varphi'_{\tilde{U}_{\Sigma_1,n}}$ die von $\tilde{U}$ berechnete Zahlfunktion, von der wir bereits festgestellt haben, dass sie n-universell für F(REK) ist.

Wenden wir die (im Beweis des Äquivalenzsatzes eingeführte) Gödelisierung von TMs auf $\tilde{U}$ an, so erhalten wir das primitiv rekursive Prädikat

$$\text{rechnung}_{\tilde{U}}^{(n+3)},$$

für das

$$\varphi(e, \vec{x}) = (\mu y(\text{rechnung}_{\tilde{U}}(e, \vec{x}, (y)_1, (y)_2)))_2$$

für alle $e \in \mathbb{N}$ und $\vec{x} \in \mathbb{N}^n$ gilt. Setzen wir also

$$T(e, \vec{x}, y) \;\Leftrightarrow\; \text{rechnung}_{\tilde{U}}(e, \vec{x}, (y)_1, (y)_2)$$

und

$$U(y) = (y)_2,$$

so ergibt sich hieraus der erste Teil des Normalformsatzes.

NORMALFORMTHEOREM VON KLEENE

SATZ: Für jede Zahl $n \geq 1$ gibt es eine primitiv rekursive Funktion $U^{(1)}$ und ein primitiv rekursives Prädikat $T^{(n+2)}$, so dass es zu jeder n-stelligen partiell rekursiven Funktion $\psi^{(n)}$ eine Zahl $e \in \mathbb{N}$ gibt mit

$$\forall \vec{x} \in \mathbb{N}^n (\psi(\vec{x}) = U(\mu y T(e, \vec{x}, y))).$$

Weiter ist die durch

$$\varphi(e, \vec{x}) = U(\mu y T(e, \vec{x}, y))$$

definierte partielle Funktion $\varphi^{(n+1)}$ eine Gödelnummerierung der n-stelligen partiell rekursiven Funktionen. Darüberhinaus können die Übersetzungsfunktionen nach φ primitiv rekursiv und injektiv gewählt werden.

Zu zeigen bleibt die Existenz injektiver, primitiv rekursiver Übersetzungsfunktionen nach φ:

Gegeben: $\hat{\varphi}^{(n+1)} \in$ F(REK).

Gesucht: $h \in$ F(PRIM) injektiv mit $\hat{\varphi}_e = \varphi_{h(e)}$ für alle $e \in \mathbb{N}$.

Idee zur Definition von h:

1. Wähle eine normierte TM M mit Programm δ, die $\hat{\varphi}$ berechnet.

2. Für jedes e gewinne aus M eine normierte TM M_e mit Programm δ_e, die bei Eingabe $\vec{x} \in \mathbb{N}^n$ die Ausgabe $\hat{\varphi}(e, \vec{x})$ wie folgt berechnet: M_e ergänzt die Eingabe $\vec{x}$ um den Parameter e und simuliert dann δ auf der erweiterten Eingabe $(e, \vec{x})$.

3. Zeige, dass sich die Gödelnummer $h(e) := gn(M_e)$ von M_e (d.h. $0^{h(e)+1}$ ist die Unärdarstellung von $\lceil \delta_e \rceil$) primitiv rekursiv berechnen lässt.

Es folgt

$$\hat{\varphi}(e, \vec{x}) = U(\mu y(T(gn(M_e), \vec{x}, y))) = \varphi(h(e), \vec{x}),$$

weshalb h die gesuchte Übersetzungsfunktion von $\hat{\varphi}$ nach φ ist. *q.e.d.*

NOTATION.

Für die Standardaufzählung $\varphi^{(n+1)}$ der n-stelligen partiell rekursiven Funktionen benutzen wir folgende Notation:

- $\{e\}^{(n)}(\vec{x}) := \varphi_e(\vec{x})$

  Sprechweise: $\{e\}^{(n)}$ ist die e-te n-st. part. rek. Funktion
  Gilt $\psi = \{e\}$, so heisst e *(part. rek.) Index* von ψ.

- $W_e^{(n)} := Db(\varphi_e) = Db(\{e\}) = \{\vec{x} : (e, \vec{x}) \in W^{(n+1)}\}$

  $W^{(n+1)} := Db(\varphi)$

  Sprechweise: $W_e^{(n)}$ ist die e-te n-st. rek. aufzb. Menge
  Gilt $A = W_e$, so heisst e *(r.a.) Index* von A.

9.3 Negative Ergebnisse über universelle Funktionen für Teilklassen der partiell rekursiven Funktionen

UNIVERSELLE REKURSIV AUFZÄHLBARE MENGEN

Universelle Mengen definiert man entsprechend zu universellen Funktionen:

DEFINITION. Eine Menge $U \subseteq \mathbb{N}^{n+1}$ ist n-*universell* für eine Klasse C von (mehrdimenionalen) Mengen über $\mathbb{N}$, falls $U \in$ C und die Zweige $U_e = \{\vec{x} : (e, \vec{x}) \in U\}$ von U gerade die n-dimensionalen Mengen in C sind, d.h.

$$\mathrm{C}^{(n)} := \mathrm{C} \cap \{A : A \subseteq \mathbb{N}^n\} = \{U_e : e \geq 0\}.$$

SATZ. Für jede n-universelle partiell rekursive Funktion φ ist der Definitionsbereich $W = Db(\varphi)$ n-universell für die Klasse der rekursiv aufzählbaren Mengen. Insbesondere gibt es also n-universelle rekursiv aufzählbare Mengen.

BEWEIS. Trivial.

Zum Abschluss dieses Abschnitts wollen wir noch zeigen, dass es - im Gegensatz zu den partiell rekursiven Funktionen - keine universellen total rekursiven bzw. primitiv rekursiven Funktionen gibt.

Da die Klassen F_{tot}(REK) und F(PRIM) den Nachfolger und die Projektionen enthalten und gegen simultane Substitution abgeschlossen sind, folgt dies aus folgender allgemeineren Beobachtung:

SATZ. Sei F eine Klasse totaler Funktionen über $\mathbb{N}$, die die Nachfolgerfunktion S sowie die Projektionen U_i^n enthält und die gegen die simultane Substitution abgeschlossen ist. Dann besitzt F keine n-universelle Funktion (für jedes $n \geq 1$).

BEWEIS DURCH *DIAGONALISIERUNG* (für $n = 1$):

Widerspruchsannahme: $f^{(2)} \in$ F sei 1-universell für F.

Definiere die Funktion g durch

$$g(x) = f(x,x) + 1 = S(f(U_1^1(x), U_1^1(x))) = S(f(U_1^1, U_1^1))(x)$$

Nach Wahl von F gilt dann $g \in$ F.

g ist aber so gewählt, dass es sich für jede Zahl e vom e-ten Zweig von f an der Stelle e unterscheidet:

$$g(e) = f(e,e) + 1 = f_e(e) + 1 > f_e(e)$$

Es gilt daher $g \notin \{f_e : e \geq 0\}$. Wegen $g \in$ F *widerspricht* dies aber der 1-Universalität von f ! *q.e.d.*

Dieses Diagonalargument versagt bei partiellen Funktionen, da für undefiniertes $\varphi(x,x)$ der Wert von $\varphi(x,x) + 1$ ebenfalls undefiniert ist und daher $\varphi(x,x) = \varphi(x,x) + 1$ gilt!

Wir können jedoch gegen die 1-universelle Funktion φ für F(REK) mit folgender Fallunterscheidung diagonalisieren:

$$g(x) = \begin{cases} \varphi(x,x) & \text{falls } \varphi(x,x)\downarrow \\ 0 & \text{falls } \varphi(x,x)\uparrow. \end{cases}$$

Dann gilt $g \neq \varphi_e$ für alle e, d.h. (wegen der Universalität von φ) dass g nicht rekursiv ist.

Wegen der Abschlusseigenschaften von F(REK) können wir hieraus folgern, dass die Menge $\{x : \varphi(x,x)\downarrow\}$ NICHT REKURSIV ist.

($\rightarrow$ Algorithmische Unlösbarkeit des Halteproblems)

NICHTREKURSIVE REKURSIV AUFZÄHLBARE MENGEN

Die Beobachtung, dass die Klassen der primitiv rekursiven und total rekursiven Funktionen keine universellen Mengen besitzen, lässt sich leicht auf die Klassen der primitiv rekursiven und rekursiven Mengen übertragen (s. Übungen):

SATZ. Die Klassen der primitiv rekursiven Mengen und der rekursiven Mengen besitzen keine n-universellen Mengen (für alle $n \geq 1$).

Wegen der Existenz universeller rekursiv aufzählbarer Mengen folgt hieraus:

KOROLLAR. Es gibt rekursiv aufzählbare Mengen, die nicht rekursiv sind.

Die schon zuvor als nicht rekursiv nachgewiesene Menge $A = \{x : \varphi(x,x)\downarrow\}$ ist natürlich auch rekursiv aufzählbar, da A der Definitionsbereich der partiell rekursiven Funktion $\psi(x) = \varphi(x,x)$ ist.

Im nächsten Abschnitt werden wir das Thema der nichtrekursiven Menge ausführlich behandeln.