Universität Karlsruhe (TH) – Fakultät für Informatik
Institut für Logik, Komplexität und Deduktionssysteme
Lehrstuhl Prof. Dr. P. Deussen

Informatik III – Prof. Dr. P. Deussen im Wintersemester 2001/2002

C. Bierwald, Th. Käufl, F. Schuster http://i10www.ira.uka.de/lehre/info3-winter2001 info3@ira.uka.de

Übungsblatt 13 – Lösungsvorschläge

25.1.2002

Aufgabe 1 Rekursionssatz (Bierwald)

Beweisen Sie den folgenden Satz:

> Zu jeder totalen und berechenbaren Funktion $f\colon \mathbb{N} \to \mathbb{N}$
> gibt es ein $n \in \mathbb{N}$, sodaß $\varphi_n = \varphi_{f(n)}$.

Hinweis. Zeigen Sie dazu zunächst mit Hilfe des smn-Theorems, daß es eine totale und berechenbare Funktion $g : \mathbb{N} \to \mathbb{N}$ mit

$$\varphi_{g(i)}(j) = \begin{cases} \varphi_{\varphi_i(i)}(j) & \text{, falls } \varphi_i(i)\downarrow \\ \perp & \text{, sonst} \end{cases}$$

gibt, betrachten Sie anschließend die Komposition $f \circ g$ und untersuchen Sie $\varphi_{g(i)}(j)$ für ein geschickt gewähltes i.

Wir betrachten die Funktion $\Psi : \mathbb{N}^2 \to \mathbb{N}$ definiert durch:

$$\Psi(i, j) = \begin{cases} \varphi_{\varphi_i(i)}(j) & \text{, falls } \varphi_i(i)\downarrow \\ \perp & \text{, sonst} \end{cases}$$

Das die Funktion berechenbar ist, zeigt das folgende while-Programm:

```
begin
    X3:= Φ(X1, X1);
    X1:= Φ(X3, X2)
end
```

Also gibt es einen Index e unter den zweistelligen berechenbaren Funktionen mit $\varphi_e^{(2)} = \Psi$. Nach dem smn-Theorem gibt es eine totale und berechenbare Funktion $s_1^1 : \mathbb{N}^2 \to \mathbb{N}$, sodaß

$$\varphi_e^{(2)}(x, y) = \varphi^{(1)}_{s_1^1(e, x)}(y)$$

Damit definieren wir die gesuchte Funktion g durch $g(i) = s_1^1(e, i)$. Auch g ist total und berechenbar (begin X2:=X1; X1:=e; „Whileprogramm für s_1^1“ end) und es gilt:

$$\varphi_{g(i)}(j) = \begin{cases} \varphi_{\varphi_i(i)}(j) & \text{, falls } \varphi_i(i)\downarrow \\ \perp & \text{, sonst} \end{cases}$$

Als Komposition berechenbarer Funktionen ist auch $f \circ g \colon \mathbb{N} \to \mathbb{N}$ berechenbar, hat also einen Index m unter den berechenbaren Funktionen. Außerdem ist $f \circ g$ total, da f und g es sind. Wir betrachten nun $\varphi_{g(m)}(j)$

$$\begin{aligned} \rho_{g(m)}(j) &= \begin{cases} \varphi_{\varphi_m(m)}(j) & \text{, falls } \varphi_m(m)\downarrow \\ \perp & \text{, sonst} \end{cases} \\ &= \begin{cases} \varphi_{f \circ g(m)}(j) & \text{, falls } f \circ g(m)\downarrow \\ \perp & \text{, sonst} \end{cases} \\ &= \varphi_{f \circ g(m)}(j) \text{ , da } f \circ g \text{ total} \\ &= \varphi_{f(g(m))}(j) \end{aligned}$$

und stellen fest, daß $\varphi_n = \varphi_{f(n)}$, falls wir $n = g(m)$ wählen.

Aufgabe 2 Seltsame selbstbezügliche While-Programme und Funktionen (Bierwald)

Ein while-Programm, das stets terminiert und als Ausgabe seinen eigenen Index liefert, wollen wir *egozentrisch* nennen, und eine berechenbare Funktion $\varphi_n \colon \mathbb{N} \to \mathbb{N}$, die an der Stelle n undefiniert ist und überall sonst den Wert 0 hat, heiße *Rumpelstilzchen*-Funktion. Zeigen Sie, daß es egozentrische while-Programme und Rumpelstilzchen-Funktionen gibt.

Hinweis. Wenden Sie den Rekursionssatz an.

Egozentrische while-Programme.

Wir betrachten die Funktion $\Psi \colon \mathbb{N}^2 \to \mathbb{N}$ definiert durch $\Psi(x, y) = x$. Diese Funktion ist offenbar berechenbar (begin end). Also gibt es einen Index e unter den zweistelligen berechenbaren Funktionen mit $\varphi_e^{(2)} = \Psi$. Nach dem smn-Theorem gibt es eine totale und berechenbare Funktion $s_1^1 \colon \mathbb{N}^2 \to \mathbb{N}$, sodaß

$$\varphi_e^{(2)}(x, y) = \varphi_{s_1^1(e, x)}^{(1)}(y).$$

Wir definieren eine neue Funktion $f \colon \mathbb{N} \to \mathbb{N}$ durch $f(x) = s_1^1(e, x)$. Dann ist f total und berechenbar und es gilt:

$$\varphi_{f(x)}(y) = \varphi_{s_1^1(e, x)}^{(1)}(y) = \varphi_e^{(2)}(x, y) = \Psi(x, y) = x$$

Nach dem Rekursionssatz gibt es ein n, sodaß $\varphi_n = \varphi_{f(n)}$, also $\varphi_n(y) = \varphi_{f(n)}(y) = n$. Damit ist das while-Programm P_n egozentrisch.

Rumpelstilzchen-Funktionen.

Wir betrachten die Funktion $\Psi : \mathbb{N}^2 \to \mathbb{N}$ definiert durch:

$$\Psi(x, y) = \begin{cases} \perp & \text{, falls } x = y \\ 0 & \text{, sonst} \end{cases}$$

Diese Funktion ist offenbar berechenbar:

begin while X1=X2 do begin end; X1:=0 end

Also gibt es einen Index e unter den zweistelligen berechenbaren Funktionen mit $\varphi_e^{(2)} = \Psi$. Nach dem smn-Theorem gibt es eine totale und berechenbare Funktion $s_1^1 : \mathbb{N}^2 \to \mathbb{N}$, sodaß

$$\varphi_e^{(2)}(x, y) = \varphi_{s_1^1(e, x)}^{(1)}(y)$$

Wir definieren eine neue Funktion f: $\mathbb{N} \to \mathbb{N}$ durch $f(x) = s_1^1(e, x)$. Dann ist f total und berechenbar und es gilt:

$$\varphi_{f(x)}(y) = \varphi_{s_1^1(e, x)}^{(1)}(y) = \varphi_e^{(2)}(x, y) = \Psi(x, y) = \begin{cases} \perp & \text{, falls } x = y \\ 0 & \text{, sonst} \end{cases}$$

Nach dem Rekursionssatz gibt es ein n, sodaß $\varphi_n = \varphi_{f(n)}$, also

$$\varphi_n(y) = \varphi_{f(n)}(y) = \begin{cases} \perp & \text{, falls } n = y \\ 0 & \text{, sonst} \end{cases}$$

Die Funktion φ_n ist eine Rumpelstilzchen-Funktion.

Aufgabe 3 Turingmaschinen (Käufl)

a) Gegeben sei eine Turingmaschine mit den Befehlen
(s, ƀ, ƀ, rechts, q_1), (q_1, |, |, rechts, q_1), (q_1, ƀ, |, links, q_2), (q_2, |, |, links, q_2),
(q_2, ƀ, ƀ, rechts, q_3), (q_3, |, ƀ, rechts, q_4), (q_3, ƀ, ƀ, rechts, q_4), (q_4, |, ƀ, stehen, h) und
(q_4, ƀ, ƀ, stehen, q_4).
Was bewirkt diese Maschine, wenn sie in der Konfiguration ƀsƀ| | ƀ | | | ƀ gestartet wird. Welche Funktion wird berechnet, wenn sie in Konfigurationen der Form ƀsƀ $|^n$ƀ $|^m$ƀ gestartet wird?

Die Turingmaschine ersetzt ƀ zwischen | | und | | | ƀ durch |. Anschließend löscht sie die beiden links stehenden Zeichen |.

Angesetzt in der Konfiguration ƀsƀ $|^n$ƀ $|^m$ƀ endet die Turingmaschine in der Konfiguration ƀhƀ $|^{n+m-1}$ƀ, berechnet also die Addition der Zahlen k und l sofern $k + 1 = n$ und $l + 1 = m$.

b) Zwei (nicht notwendig verschiedene) Turingmaschinen mit den Befehlen B_1, B_2, den Startzuständen s_1, s_2 und den Haltezuständen h_1 und h_2 können zu einer neuen zusammengesetzt werden. Die Befehlsmenge B ist die Vereinigung von B_1 und B_2, wobei in B_1 mindestens ein Vorkommnis von h_1 durch s_2 ersetzt wird.

Betrachten Sie die Turingmaschine r mit den Befehlen (s, ƀ, ƀ, rechts, h), (s, |, |, rechts, h), (h, stop) und die Zusammensetzung R, die Sie durch Zusammensetzung von r mit sich selber erhalten, wobei (s, |, |, rechts, h) durch (s, |, |, rechts, s) ersetzt wird. Was bewirken r und R?

Diagramm für R: |

r $\xrightarrow{\text{ƀ}}$

r geht um ein Feld nach rechts. R geht um ein Feld nach rechts. Solange das neue betrachtete Feld das Zeichen | enthält, geht R wieder um ein Feld nach rechts, Sobald das betrachtete Feld ƀ enthält. bleibt R stehen. Weder r und R noch ändern den Bandinhalt.

c) Definieren Sie Turingmaschinen, die an der betrachteten Position ƀ bzw. | eintragen und eine Turingmaschine, die von der betrachteten Position aus solange nach links läuft, bis sie ƀ vom Band liest.

b (ƀ-Maschine): (s, ƀ, ƀ, stehen, h), (s, |, ƀ, stehen, h), (h, stop)

1 (| -Maschine): (s, ƀ, |, stehen, h), (s, |, |, stehen, h), (h, stop)

l (kleine Linksmaschine): (s, ƀ, ƀ, links, h), (s, |, |, links, h), (h, stop)

L (große Linksmaschine): In der Befehlsmenge (s, |, |, links, h) durch (s, |, |, links, s) ersetzen.

d) Geben Sie eine Turingmaschine an, die die Addition von zwei Zahlen ausführt. Verwenden Sie dabei eine geeignete Komposition der in Teilaufgabe b und c definierten Maschinen.

Additionsmaschine: R1Lrƀ rƀ

e) Geben Sie eine Turingmaschine (Kopiermaschine) an, die in der Startkonfiguration ein leeres Feld beobachtet und gestartet die Konfiguration ƀsƀ $|^n$ƀ in ƀsƀ $|^n$ƀ $|^n$ƀ überführt. Die Maschine kopiert also die rechts vom beobachteten Feld stehende Strichfolge.

Konstruktionsprinzip. Die Maschine ersetzt einen Strich der gegebenen Strichfolge durch ƀ, läuft nach rechts (zweimalige Anwendung von R) und schreibt in das Feld einen |. Anschließend läuft sie zum linken Ende der Kopie und geht um ein weiteres Feld nach links. Enthält dieses ein ƀ ist die Strichfolge kopiert. Im anderen Fall ist weiter zu kopieren.

|

r bRR 1 LL 1 r $\xrightarrow[\text{ƀ}]{}$

Sind M_1 und M_2 zwei Turingmaschinen, dann besagt $\mathsf{M}_1\mathsf{M}_2$, daß in der Befehlsmenge von M_1 der Haltezustand h_1 durch den Startzustand s_2 von M_2 zu ersetzen ist. Bei der kleinen Rechtsmaschine (im Diagramm der Kopiermaschine) ist in dem Befehl, bei dem | das beobachtete Zeichen ist, der Haltezustand durch den Startzustand der ƀ-Maschine zu ersetzen und im Befehl mit ƀ als beobachteten Feld ist der Haltezustand durch den Startzustand der kleinen Linksmaschinel zu ersetzen.

f) Geben Sie eine Turingmaschine an, die die Strichfolgen $|^n$, n = 1, 2, … jeweils getrennt durch ƀ auf ein sonst leeres Band schreibt.

Dabei ist K die in e definierte Kopiermaschine.

1 l KrRR1L

Abgabe der Aufgabe 1 zur Korrektur bei Ihrem Tutor in der Zeit vom 28.1.–1.2.2002.