LÖSUNG VON:
David Luescher (luescdav) und Pascal Brunner (brunnpa7)

Zürcher Hochschule
für Angewandte Wissenschaften

Theoretische Informatik
D. Flumini, L. Keller, O. Stern

Übungsblatt 8

Berechenbarkeit

Lösungen sind unten aufgeführt

Abgabe: Kalenderwoche 20

Aufgabe 1.
Geben Sie LOOP-Programme an, die folgende Funktionen berechnen (ohne Verwendung von Makros).

(a) $f\colon \mathbb{N}^2 \to \mathbb{N}$ mit $f(a,b) = \max(a,b)$

(b) $h\colon \mathbb{N}^2 \to \mathbb{N}$ mit $h(x,y) = |x-y|$

(c) $i\colon \mathbb{N} \to \mathbb{N}$ mit

$$i(x,y) = \begin{cases} \max(x-y,0) & \text{falls } x \text{ ungerade} \\ \max(y-x,0) & \text{falls } x \text{ gerade} \end{cases}$$

15 Punkte

Aufgabe 2.
Geben Sie WHILE-Programme an, die folgende Funktionen abbilden (ohne Verwendung von Makros).

(a) $f(n) = \begin{cases} 0 & \text{falls } (2n)^2 > 0 \\ \uparrow & \text{sonst} \end{cases}, n \in \mathbb{N}$

zuerst überlegen was für ein Output erwartet wird. Die Implementierung des LOOP-Programms muss nicht 1:1 identisch sein mit der Aufgabenstellung, der Output muss einfach für jeden Fall stimmen

(b) $g(n) = \begin{cases} n + g(n-1) & \text{falls } n > 0 \\ \uparrow & \text{sonst} \end{cases}, n \in \mathbb{N}$

(c) $h(n) = n * (\log_2(4 * \max(4^n, \log_2(n+1))) - n - 2), n \in \mathbb{N}$

Hinweis: Versuchen Sie zuerst die Funktionen soweit wie möglich zu vereinfachen, bevor Sie mit der effektiven Implementierung beginnen.

15 Punkte

Aufgabe 3.
Seien $A, B \subseteq \Sigma^*$ zwei beliebige Sprachen. Beweisen Sie folgende Aussagen.

(a) Wenn A und B beide entscheidbar sind, dann ist auch $A \cap B$ entscheidbar.

(b) Wenn A entscheidbar ist, dann ist auch das Komplement $\overline{A}$ entscheidbar.

(c) Wenn A und B beide semi-entscheidbar sind, dann ist auch $A \cup B$ semi-entscheidbar.

Hinweis: Sie können die Beweise mithilfe von Pseudocode und falls notwendig mit einer kleinen Beschreibung durchführen. **15 Punkte**

Aufgabe 1.
Geben Sie LOOP-Programme an, die folgende Funktionen berechnen (ohne Verwendung von Makros).

(a) $f\colon \mathbb{N}^2 \to \mathbb{N}$ mit $f(a,b) = \max(a,b)$

(b) $h\colon \mathbb{N}^2 \to \mathbb{N}$ mit $h(x,y) = |x-y|$

(c) $i\colon \mathbb{N} \to \mathbb{N}$ mit

$$i(x,y) = \begin{cases} \max(x-y,0) & \text{falls } x \text{ ungerade} \\ \max(y-x,0) & \text{falls } x \text{ gerade} \end{cases}$$

15 Punkte

a)
```
x3 = x1 - x2;
x3 = 1 - x3;
LOOP x3 DO
        x0 = x2 + 0;
END;
x3 = 1 - x3;
LOOP x3 DO
        x0 = x1 + 0;
END
```

b)
```
x3 = x1 - x2;
x3 = 1 - x3;
LOOP x3 DO
        x0 = x2 - x1;
END;
x3 = 1 - x3;
LOOP x3 DO
        x0 = x1 - x2;
END
```

c)
```
x3 = x1 + 0;
LOOP x1 DO
        x1 = x1 - 2;
        x4 = 1 - x1;
        x4 = 1 - x4;
        LOOP x4 DO
                x3 = x1 + 0;
        END;
END;
x1 = 2 - 1;
x1 = x3 - x1;
LOOP x1 DO
        x3 = 0 + 0;
END;
x5 = 1 - x3;
LOOP x5 DO
        x0 = x2 - x1;
END;
x5 = 1 - x5;
LOOP x5 DO
        x0 = x1 - x2;
END
```

Aufgabe 2.
Geben Sie WHILE-Programme an, die folgende Funktionen abbilden (ohne Verwendung von Makros).

(a) $f(n) = \begin{cases} 0 & \text{falls } (2n)^2 > 0 \\ \uparrow & \text{sonst} \end{cases}, n \in \mathbb{N}$

(b) $g(n) = \begin{cases} n + g(n-1) & \text{falls } n > 0 \\ \uparrow & \text{sonst} \end{cases}, n \in \mathbb{N}$

(c) $h(n) = n * (\log_2(4 * \max(4^n, \log_2(n+1))) - n - 2), n \in \mathbb{N}$

zuerst überlegen was für ein Output erwartet wird. Die Implementierung des LOOP-Programms muss nicht 1:1 identisch sein mit der Aufgabenstellung, der Output muss einfach für jeden Fall stimmen

Hinweis: Versuchen Sie zuerst die Funktionen soweit wie möglich zu vereinfachen, bevor Sie mit der effektiven Implementierung beginnen. **15 Punkte**

a) im Wesentlichen gibt es zwei Fälle. Entweder ergibt die Berechnung 0 oder >0. Wenn >0 liefere 0 zurück. Wenn =0 dann bleib in der Endlosschlaufe

```
x0 = 1 - x1;
WHILE x0 > 0 DO
END
```

b) Es gibt folgende Fälle:
n = 0 -> Programm terminiert nicht
n = 1 -> 1 + g(0) -> terminiert nicht
n > 1 -> Rekursive Funktion -> terminiert nicht
=> Es ist ein Programm, dass nie terminiert

```
x0 = 1 + 0;
WHILE x0 > 0 DO
END
```

c)
n*(log2(4*4^n)-n-2)
= n*(log2(4^n+1)-n-2)
= n*(((n+1)*2)-n-2)
= n*(2n+2-n-2)
= n* n = n^2

```
x2 = x1 +0;
x0 = 0 + 0;
WHILE x2 > 0 DO
          LOOP x1 DO
                    x0 = x0 + x1;
          END;
          x2 = 0 + 0;
END
```

Aufgabe 3.
Seien $A, B \subseteq \Sigma^*$ zwei beliebige Sprachen. Beweisen Sie folgende Aussagen.

(a) Wenn A und B beide entscheidbar sind, dann ist auch $A \cap B$ entscheidbar.

(b) Wenn A entscheidbar ist, dann ist auch das Komplement $\overline{A}$ entscheidbar.

(c) Wenn A und B beide semi-entscheidbar sind, dann ist auch $A \cup B$ semi-entscheidbar.

Hinweis: Sie können die Beweise mithilfe von Pseudocode und falls notwendig mit einer kleinen Beschreibung durchführen. **15 Punkte**

a)
Wenn A und B entscheidbar ist, dann ist auch der Durchschnitt von A und B entscheidbar. Für das Entscheidungsverfahren für die Menge A gibt es die TM A und für das Entscheidungsverfahren für die Menge B gibt es die TM B. Nun können wir eine TM C bauen, welche das Entscheidungsverfahren von der Menge A und der Menge B zusammenfasst. Dadurch kann für einen gegebenen Input w entschieden werden, ob es zum Durchschnitt der Menge A und B gehört.

```
PSEUDOCODE:
input (w)
IF (w € A)
        IF (w € B)
                Return 1;
        ELSE Return 0;
ELSE Return 0;
```

b)
Wenn A entscheidbar ist, dann existiert eine TM M die das Entscheidungsproblem (∑ , A) löst, indem gilt:

x € A => 1
x !€ A => 0

Wir bauen eine TM N, so dass

x € A => 0
x !€ A => 1

```
PSEUDOCODE:
IF x € A
        Return 0;
ELSE
        Return 1;
```

€ = "Element von"
!€ = "nicht Element von"

c)
Seien A und B semi-entscheidbar, dann gibt es TM M und TM N, so dass:

€ = "Element von"
!€ = "nicht Element von"

TM M:
w € A => 1;
w !€ A => terminiert nicht

TM N:
w € B => 1;
w !€ B => terminiert nicht

Wir setzen die TM M und TM N zur TM P zusammen

PSEUDOCODE:

```
input (w)
IF w € A
        Return 1;
ELSE IF w € B
        Return 1;
ELSE
        While (1 > 0) DO
        End
```