Eidgenössische Technische Hochschule Zürich
Ecole polytechnique fédérale de Zurich
Politecnico federale di Zurigo
Swiss Federal Institute of Technology Zurich

Institut für Theoretische Informatik
Dr. B. Gärtner, Prof. Dr. J. Hromkovič

13. November 2007

Informatik für Mathematiker und Physiker — Lösung 7 — HS 07

URL: http://www.ti.inf.ethz.ch/ew/courses/Info1_07/

Aufgabe 1 (Schnellübung). Das folgende Programm berechnet die Approximation der Quadratwurzel einer positiven reellen Zahl s.

Programm: babylonian.C

```
// Program: babylonian.C
// Approximation of the square root of a positive real number

#include <iostream>

int main(){

    // Read input
    double s; // input number
    double eps = 0.01; // the epsilon, i.e. max square error

    std::cout << "Which number do you want to take the square root of?";
    std::cout << "\n";
    std::cin >> s;

    // Compute square root
    double x = s / 2.0; // initialize solution
    unsigned int n = 0; // counter for number of iterations

    while (x * x - s >= eps || s - x * x >= eps) {
        ++n;
        x = (x + s / x) / 2.0;
    }

    std::cout << "The square root of " << s << " is: " << x << std::endl;
    std::cout << "The number of iterations done: " << n << std::endl;

    return 0;
}
```

Bemerkungen:
Der Startwert ist $x_0 = \frac{s}{2}$. Das ist gut für Zahlen $s > 2$, hingegen nicht so gut für Zahlen $s < 2$.

Obwohl es in der Aufgabenstellung nicht gefragt war, gibt `babylonian.C` auch noch die Anzahl der Iterationen aus, die durchgeführt wurden. Die Initialisierung zählt dabei nicht mit.

Skript-Aufgabe 3.10
Lösung in separatem Anhang zum Buch. Siehe Internetseite des Kurses.

Skript-Aufgabe 3.21
Für alle Elemente auf der Diagonalen wählen wir eine andere Ziffer aus $\{0,1,2,3,4,5,6,7,8\}$. Also zum Beispiel $c = 0.1638783\ldots$.

Skript-Aufgabe 4.3
Sei $\mathcal{F} \dot\cup \mathcal{I} = \mathbb{R}$ die Partition der reellen Zahlen in diejenigen mit endlicher Repräsentation ($\mathcal{F}$) und diejenigen ohne endliche Repräsentation ($\mathcal{I}$). Wir wollen zeigen, dass $|\mathcal{F}| \ll |\mathcal{I}|$. Aus der Vorlesung wissen Sie, dass eine Menge, deren Elemente alle eine endliche Darstellung haben, die gleiche Kardinalität wie $\mathbb{N}$ hat. Insbesondere gibt es also eine Bijektion $f : \mathcal{F} \to \mathbb{N}$, und somit $|\mathcal{F}| = |\mathbb{N}|$.

Um einen Beweis durch Widerspruch zu führen, nehmen Sie an, dass es eine bijektive Funktion, $m : \mathcal{F} \to \mathcal{I}$, gibt, die jedem Element von $\mathcal{F}$ ein eindeutiges Element von $\mathcal{I}$ zuordnet, und umgekehrt. Dann würde gelten, dass $|\mathcal{F}| = |\mathcal{I}|$; die beiden Mengen wären gleich gross.

In diesem Fall können wir eine weitere bijektive Funktion, $g : \mathbb{R} \to \mathbb{N}$, wie folgt definieren:

$$g(r) := \begin{cases} 2 \cdot f(r) & \text{falls } r \in \mathcal{F}, \\ 2 \cdot f(m^{-1}(r)) - 1 & \text{falls } r \in \mathcal{I}. \end{cases}$$

Wir ersparen uns den Beweis, dass g tatsächlich eine Bijektion zwischen $\mathbb{R}$ und $\mathbb{N}$ ist. Die wichtige Konsequenz ist, dass $|\mathbb{R}| = |\mathbb{N}|$ gelten würde. Sie wissen natürlich aus der Vorlesung, dass $|\mathbb{R}| \gg |\mathbb{N}|$. Da haben wir unseren Widerspruch, und somit muss $|\mathcal{F}| \neq |\mathcal{I}|$ gelten. Desweiteren ist es nicht schwer zu sehen, dass $|\mathcal{F}| \ll |\mathcal{I}|$, und nicht umgekehrt.

Skript-Aufgabe 4.8
Das Entscheidungsproblem $(\mathbb{N}, M(2-\mathrm{DIAG}))$ ist nicht algorithmisch lösbar, da es kein Programm gibt, das $M(2-\mathrm{DIAG})$ akzeptiert. Erinnern Sie, dass jeder Entscheidungsalgorithmus dieser Form auch als Programm geschrieben werden kann. Deshalb gibt es auch keinen Algorithmus, der obiges Problem entscheidet.

Die folgende Tabelle deutet an, dass $M(2-\mathrm{DIAG})$ sich in mindestens einem Element von der akzeptierten Mengen eines beliebigen Programmes unterscheidet. Für Programm P_i ist das natürlich genau das Element $2i$.

	1	2	3	4	5	6	7	8	…
$M(P_1)$	0	$\boxed{1}$	0	1	0	0	0	1	…
$M(P_2)$	1	1	0	$\boxed{0}$	1	0	1	1	…
$M(P_3)$	1	0	1	0	1	$\boxed{0}$	0	1	…
$M(P_4)$	1	1	1	1	1	1	1	$\boxed{0}$	…
⋮									
$M(2-\mathrm{DIAG})$	0	0	0	1	0	1	0	1	…