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

6. November 2007

Informatik für Mathematiker und Physiker Serie 7 HS 07

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

Aufgabe 1 [Schnellübung – 20 Min.]

Implementieren Sie die sogenannte Babylonische Methode zur iterativen Berechnung der Quadratwurzel einer positiven Zahl $s \in \mathbb{R}^+$. Dies funktioniert folgendermassen.

Die n-te Approximation von $x = \sqrt{s}$ wird durch die folgende rekursive Vorschrift bestimmt,

$$x_n = \frac{1}{2}(x_{n-1} + \frac{s}{x_{n-1}}).$$

Als Startwert x_0 kann im Prinzip ein beliebiger Wert genommen werden, aber es ist für die Konvergenz von Vorteil wenn $x_0 \approx x$.

Es gilt

$$\lim_{n \to \infty} x_n = x.$$

In dieser Aufgabe sollen Sie die Iteration beenden, sobald das Quadrat der Approximation nahe genug an s heran kommt. Konkret heisst das, dass Sie aufhören können, sobald die i-te Approximation folgendes erfüllt:

$$|x_i^2 - s| < 0.01.$$

Schreiben sie ein C++ Programm `babylonian.C`, das s einliest und das erste x_i ausgibt, für welches die Abbruchbedingung erfüllt ist. Wählen sie dazu den Startwert x_0 und berechnen sie mit einer Schleife die Folge der Approximationen x_i.

Hinweise:

- Ihre Abgabe sollte ein vollständiges Programm sein, das man ohne weiteres übersetzen und laufen lassen könnte.
- Verwenden Sie zur Berechnung einen geeigneten C++ Datentyp. Sie müssen sich dabei nicht um mögliche Verletzungen des Wertebereiches kümmern.
- Sie dürfen bei der Implementierung keine Bibliotheksfunktionen verwenden, ausgenommen natürlich des Gebrauches von `std::cin` und `std::cout`. Das bedeutet keine Verwendung von `abs()` oder `sqrt()`.

Die folgenden Aufgaben sind aus den Kapiteln 3 und 4 des Buches “Sieben Wunder der Informatik” entnommen. Sie finden die entsprechenden Kapitel auf der Internetseite des Kurses unter Material. Bitte beachten Sie, dass in dem Buch gewisse Aufgaben als “challenge” ausgewiesen werden. Das hat nichts mit den herausfordernden Aufgaben zu tun, die wir Ihnen üblicherweise zur Auswahl geben. Diese Woche fällt dies aus.

Skript-Aufgabe 3.10 (4 Punkte)

Assume that Hotel Hilbert is empty, i.e. there is no guest accomodated in the hotel. Since all used accomodation strategies were based on moving former guests from one room to another, there is the risk that staying in the hotel becomes unpopular. Therefore, the porter needs an accomodation strategy that does not require any move of an already accomodated guest. This accomodation strategy has to work even if arbitrarily many finite and infinite buses arrive in arbitrarily many different monents. Can you help the porter?

Skript-Aufgabe 3.21 (4 Punkte)

Let $c \in [0, 1]$ be a number that is not represented in the hypothetic listing of all those real numbers, see the figure below. Propose a choice for the first 7 digits of c behind the decimal point. Note that, c is not uniquely determined by the table.

	0	1	2	3	4	5	6	7	...
1	0.	$\boxed{2}$	0	0	1	7	8	0	...
2	0.	1	$\boxed{7}$	3	1	7	8	4	...
3	0.	1	6	$\boxed{4}$	3	3	3	3	...
4	0.	1	6	3	$\boxed{0}$	7	8	4	...
5	0.	1	6	3	1	$\boxed{8}$	8	4	...
6	0.	1	6	3	1	7	$\boxed{9}$	4	...
7	0.	1	6	3	1	7	8	$\boxed{4}$	...
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$

Skript-Aufgabe 4.3 (4 Punkte)

What do you think? Are there more real numbers with finite representation than real numbers without any finite representation, or vice versa? Justify your answer!

Skript-Aufgabe 4.8 (4 Punkte)

Consider that

$$M(2\text{-DIAG}) = \text{the set of all even numbers } 2i\text{, such that } 2i \text{ is not in } M(P_i).$$

Is the decision problem $(\mathbb{N}, M(2\text{-DIAG}))$ algorithmically solvable or not? Explain carefully your argument! Draw also pictures that would show the construction of 2-DIAG.

Abgabe: Bis 13. November 2007, 15.15 Uhr.