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

1. November 2011

Informatik für Mathematiker und Physiker Serie 6 HS 11

URL: http://www.ti.inf.ethz.ch/ew/courses/Info1_11/

Skript-Aufgabe 58 (4 Punkte)

Evaluate the following expressions step-by-step, according to the conversion rules of mixed expressions. We assume a floating point representation according to IEEE 754, that is, `float` corresponds to $F^*(2, 24, -126, 127)$ and `double` corresponds to $F^*(2, 53, -1022, 1023)$. We also assume that 32 bits are used to represent `int` values.

a) `6 / 4 * 2.0f - 3`

b) `2 + 15.0e7f - 3 / 2.0 * 1.0e8`

c) `392593 * 2735.0f - 8192 * 131072 + 1.0`

d) `16 * (0.2f + 262144 - 262144.0)`

Skript-Aufgabe 68 (4 Punkte)

The number π can be defined through various infinite sums. Here are two of them.

$$\begin{aligned} \frac{\pi}{4} &= 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots \\ \frac{\pi}{2} &= 1 + \frac{1}{3} + \frac{1 \cdot 2}{3 \cdot 5} + \frac{1 \cdot 2 \cdot 3}{3 \cdot 5 \cdot 7} + \cdots \end{aligned}$$

Write a program for computing an approximation of π, based on these formulas. Which formula is better for that purpose?

Skript-Aufgabe 70 (4 Punkte)

There is a well-known iterative procedure (the *Babylonian method*) for computing the square root of a positive real number s. Starting from any value $x_0 > 0$, we compute a sequence $x_0, x_1, x_2, \ldots$ of values according to the formula

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

It can be shown that

$$\lim_{n \to \infty} x_n = \sqrt{s}.$$

Write a program `babylonian.cpp` that reads in the number s and computes an approximation of $\sqrt{s}$ using the Babylonian method. To be concrete, the program should output the first number x_i such that

$$|x_i^2 - s| < 0.001.$$

Skript-Aufgabe 73 (4 Punkte)

We have seen that there are decimal numbers without a finite binary representation (such as 1.1 and 0.1). Conversely, every (fractional) binary number *does* have a finite decimal representation, a fact that may be somewhat surprising at first sight. Prove this fact!

More formally, given a number b of the form

$$b = \sum_{i=1}^{k} b_i 2^{-i}, \quad b_1, b_2, \ldots, b_k \in \{0, 1\},$$

prove that there is a natural number ℓ such that b can be written as an ℓ-digit decimal number

$$b = \sum_{i=1}^{\ell} d_i 10^{-i}, \quad d_1, d_2, \ldots, d_\ell \in \{0, 1, \ldots, 9\}.$$

Die **Aufgaben 53** und **75** sind die **Challenge Aufgaben** und geben jeweils 8 Punkte.

Abgabe: Bis 8. November 2011, 15.15 Uhr.