Algorithms and Data Structures **Summer 2008**
K. Mehlhorn and R. Seidel **Mo. Sept 1st, morning**
Exercise 1

Motivation

In class we studied integer multiplication. Here, we study integer division and techniques for checking the correctness of a multiplication.

Exercises

1. KM was taught the following technique for checking multiplications in school. In order to check whether $a \cdot b = c$, compute the digit sums of a, b, and c, say d_a, d_b, and d_c, and check whether $d_a \cdot d_b = d_c$. If $d_a \cdot d_b \bmod 9 = d_c$, accept the result of the multiplication.

   - Show that $10^i \bmod 9 = 1$ for integer $i \geq 0$.
   - Show that the digit sum d_a of an integer a is equal to $a \bmod 9$.
   - Explain the mathematics behind the test.

2. Elferprobe, Casting out Elevens

   - Show that powers of 10 have simple reminders module 11, namely $10^i \bmod 11 = (-1)^i$ for $i \geq 0$.
   - Describe a simple test for checking the correctness of a multiplication module 11.

3. Formulate an algorithm for dividing an n-digit integer by a k-digit integer.

   Hint: Divide 10235 by 456 and formulate your actions as an algorithm.

   What is the time complexity of the algorithm?

4. The Newton-Raphson technique for finding a zero of a function $f(x)$ is as follows. Let x^* be such that $f(x^*) = 0$. We start with an approximation x_0 of x^* and then compute iteratively, $x_{i+1} = x_i - f(x_i)/f'(x_i)$ for $i \geq 0$. For a large class of functions, the sequence x_i converges to x^*, if x_0 is sufficiently good approximation of x^*.

   - Consider the tangent of f at the point $(x_i, f(x_i))$. Where does it intersect the x-axis?
   - Let D be a real number. We want to compute the inverse $1/D$ of D.
     - Consider the function $f(x) = 1/x - D$. What is the zero x^* of f?
     - Formulate the Newton-Raphson iteration for f. Show that $x_{i+1} = x_i(2 - x_i D)$.

– Let $\delta_i = x^* - x_i$. Express δ_{i+1} in terms of δ_i.
– Assume that $1 \le D \le 2$ and set $x_0 = 3/4$. Then $|\delta_0| \le 1/4$. Derive a bound on δ_i.
– Assume now that D is given as a binary fraction $D_0.D_{-1}\ldots D_{-n}$ with $D_i \in \{0,1\}$, $D_0 = 1$ and $D = \sum_{-n \le i \le 0} D_i 2^i$. We want to compute a number x such that $|x - 1/D| \le 2^{-n}$.
  We start with $x_0 = 3/4$. Estimate k such that $|x_k - 1/D| \le 2^{-n}$.
– Right or Wrong? We obtain x_{i+1} from x_i by two multiplications and one subtraction. Using Karatsuba multiplication, the time for an iteration is $O(n^{\log 3})$. Since only a logarithmic number of iterations are required, we can divide in time $O(n^{\log 3} \log n)$.
– (difficult): We modify the iteration formula as follows. We write our iterates as binary fractions, i.e., $x_i = 0.x_{i,-1}x_{i,-2}\ldots x_{i,-n}\ldots$. Then $x_0 = 0.11$. After each iteration we truncate the new iterate to $n+10$ bits after the binary point. Will the iteration still converge to $1/D$? How fast will it converge? Does the resulting method have running time $O(n^{\log 3} \log n)$?

Have fun with the solution!