Math 310 Numerical Analysis (Bueler) 29 October 2019

Worksheet: Accurate square roots

A. Without using a calculator or computer, and without expending any real effort, roughly estimate the square root of $x = 4.42341259812 \times 10^{38}$. (*Hint*: The exponent is even.)

B. Recall the form that a positive number x would have if it were written in IEEE floating point representation,

$$\begin{aligned} x &= \begin{array}{|c|c|c|c|c|c|c|c|c|c|}\hline 0 & e_1 & e_2 & \dots & e_{11} & b_1 & b_2 & b_3 & \dots & b_{52} \\ \hline \end{array} \\ &= +\,(1.b_1b_2b_3\dots b_{52})_2 \,\times\, 2^{(e_1e_2\dots e_{11})_2 - 1023_{10}} \end{aligned}$$

But a simpler view is that the parts are a real number $1 \le v < 2$ and an integer k, and that they represent x:

$$x = v \times 2^k \tag{$*$}$$

So now do a binary version of **A** by using the idea $(*)$. For example, roughly estimate the square root of $x = 1.110001101_2 \times 2^{10110_2} = v \times 2^k$. (*Hint*: The exponent is even.) Write the answer in the form $w \times 2^m$ where $1 \le w < 1$ and m is an integer.

C. What if the exponent is odd? Try $x = 1.110001101_2 \times 2^{10111_2}$.

D. Brainstorm with your group on this part. Again suppose the positive number x has expression $x = v \times 2^k$ where $1 \le v < 2$ and k is an integer. But also assume k is even for this part. Using form $(*)$, decide how to *quickly* compute an accurate approximation of $\sqrt{x}$, say with 14 decimal digit accuracy, using only the elementary operations of addition, subtraction, multiplication, and division. Note that we have several technologies for approximating, namely equation solvers (bisection, Newton, and secant) and also polynomial approximation schemes (Taylor and interpolation).

I have written and posted online a short code that extracts the v and k in $(*)$. You do not need to know how it is implemented.

```
function [v,k,s] = ieeeparts(x);
% IEEEPARTS  Every floating point (double) number x has internal
% representation
%   x = (-1)^s (1.b1 b2 b3 ... b52)_2 x 2^{(e1 e2 ... e11)_2 - 1023)
%     = (-1)^s v x 2^k
% This function returns these parts.  Thus for any real number x
%   [v,k,s] = ieeeparts(x)
% satisfies
%   v = real number in the interval  [1,2)
%   k = integer in the list  {-1022,...,1023}
%   s = 1 if x is negative and 0 otherwise
```

E. Using the result of brainstorming in **D**, and using `ieeeparts(x)` as shown below, fill-in this function that computes square roots using only the elementary operations. In particular, your code must exploit $x = v \times 2^k$ with k either even or odd.

```
function z = mysqrt(x)
% MYSQRT  Computes the square root z of a positive input x.  Compare SQRT.

if x < 0
    error(’MYSQRT only works for x >= 0’)
end
if x == 0
    z = 0;
    return
end

[v,k] = ieeeparts(x);
```