Numerical Analysis I Paper 3 Question 6 2005 MROD

Solution Notes

Context: IEEE arithmetic; machine epsilon.

(a) A floating-point number may be represented in the form

$$\pm d_0 . d_1 d_2 \ldots d_{p-1} \times \beta^e$$

where the p significant digits of base β, $d_0 . d_1 \ldots d_{p-1}$, form the significand. The sign bit represents the sign of the significand as 0 (positive) or 1 (negative). The exponent e is characterised by its minimum and maximum values, e_{min} and e_{max}. A normalised number is in the unique form such that $d_0 \neq 0$. A denormal number is a number that can be represented with exponent e_{min} but which is too small to be normalised. Denormal numbers are conventionally stored with exponent $e_{min} - 1$.
In IEEE binary arithmetic the value of d_0 can be deduced from the exponent, so is not stored. This is called the hidden bit and is 1, for normalised numbers, or 0, for denormal numbers. The precision p is the number of digits of the significand including the hidden bit.

[7 marks]

(b) $(+\infty) \,\omega\, x$ evaluates to $+\infty$ in each case

$x + (-\infty)$ " " $-\infty$

$x - (-\infty)$ " " $+\infty$

$x * (-\infty)$ " " $-\infty$

$x \,/\, (-\infty)$ " " -0

[4 marks]

(c) e_{min} must be -510 so that there are 1024 exponent values (including $e_{min}-1$ and $e_{max}+1$) requiring 10 bits. If 1 bit is needed for the sign, then 37 bits are left for the significand, so $p = 38$ including the hidden bit.

[4 marks]

over

(d) If x^* is the floating-point representation of x, then <u>machine epsilon</u> is the smallest $\epsilon_m > 0$ such that $(1+\epsilon_m)^* > 1$. [1 mark]

(e) To avoid computing x^2, divide top and bottom by x

$$f(x) = \frac{(x+1)^2}{x^2+1} = \frac{(1+1/x)(x+1)}{x+1/x}.$$

If $(1/x) < \epsilon_m$ then $(1+1/x)^* = 1$, $(x+1/x)^* = x$, $(x+1)^* = x$, so $[f(x)]^* = 1$. [4 marks]