exploringbinary.com

Why Powers of Ten Up to 1022 Are Exact As Doubles

Aug. 11th, 2011

Send to Kindle

The fast path in David Gay's decimal to floating-point conversion routine relies on this property of the first twenty-three nonnegative powers of ten: they have exact representations in double-precision floating-point. While it's easy to see why powers of ten up to 10^{15} are exact, it's less clear why the powers of ten from 10^{16} to 10^{22} are. To see why, you have to look at their binary representations.

Trailing Zeros Become "Significantly Insignificant"

10^{15} is the largest power of ten less than $2^{53} - 1$, the largest integer representable in 53 bits. 10^{15} has 50 bits, and the next larger power of ten, 10^{16}, has 54 bits. So how can 10^{16} — and the 74-bit 10^{22} for that matter — have an exact double-precision representation, when doubles only store a 53-bit significand? The answer is simple: it has a binary representation with trailing zeros after bit 53.

A nonnegative power of ten 10^n equals $5^n * 2^n$; in binary, that's a power of five followed by n zeros (a power of five always ends in '1'). For example, 10^{22} (10000000000000000000000) is

```
10000111100001100111100000110010011011101010110010010000000000000000
```

This is 5^{22} (2384185791015625), which is 52 bits long, followed by 22 zeros. Here it is in binary scientific notation, showing only its significant bits — the bits of its power of five factor:

1.000011110000110011110000011001001101110101011001001 x 2^{73}

Since this is 52 bits, it maps directly to a double — no rounding is required.

So what happened to the 22 significant trailing zeros? They were preserved in the exponent, as the factor 2^{22}. (The remaining factor, 2^{51}, undoes the normalization.)

Why 10^{22} Is the Maximum

To see why 10^{22} is the maximum exact power of ten, look at 10^{23}. Its power of five factor is 5^{23} (11920928955078125), which is 54 bits long; this is too long for exact representation in a double.

10^{16} to 10^{22}

Here are the powers of ten from 10^{16} to 10^{22}, shown in binary and in binary scientific notation (in the binary representation, bits 54 and beyond are highlighted):

On Binary Scientific Notation for Exact Integers

In decimal scientific notation, when trailing zeros are significant, they are displayed. For example, if you know a value to be exactly 300,000, you would write it as 3.00000 x 10^5. In binary scientific notation, however, this convention is not followed. I don't know if there's an official explanation, but I have my own.

Binary scientific notation is used to represent binary floating-point numbers, and binary floating-point numbers are, in general, approximations to decimal numbers. Displaying trailing zeros, which could be the result of rounding, might imply an exactness that's not there. Also, binary floating-point numbers have a limited length significand; keeping the number of significant bits displayed within this limit makes the mapping clearer.

Other Large Numbers With Exact Representations

For the same reason, other large numbers — like 14^{18} (greater than 10^{20}), 6^{33} (greater than 10^{25}), and 2^{1023} (greater than 10^{307}) — are exact as doubles. In fact, any integer with a power of two factor that takes it beyond 53 bits can be represented exactly (as long as the maximum exponent isn't exceeded). In this case, **a double can represent more than 53 significant bits!**

EB

https://www.exploringbinary.com/why-powers-of-ten-up-to-10-to-the-22-are-exact-as-doubles/