Re: Glaring Excel calc bug.

Source: http://www.tech−archive.net/Archive/Excel/microsoft.public.excel/2007−06/msg01540.html

- *From*: "Earl Kiosterud" <someone@xxxxxxxxxxx>
- *Date*: Fri, 29 Jun 2007 13:07:16 −0400

Jerry,

I agree that neither decimal nor binary can represent values precisely. I was just referring to the conversion issues. If we have decimal values to start with, and want decimal results, why not do the math in decimal? We can certainly afford the memory and speed hits with todays hardware.
−−
Earl Kiosterud
www.smokeylake.com

_____

"Jerry W. Lewis" <post_a_reply@xxxxxxxxxxxxx> wrote in message news:6D39EFF7−0AE9−4886−A7E3−B9B1D254A18B@xxxxxxxxxxxxxxx

> Digital computers have always been inherrently binary. You can impose decimal representations on to binary patterns (BCD), but the cost is slower speed and and inefficient use of memory
> http://en.wikipedia.org/wiki/Binary_coded_decimal
>
> Using BCD does not eliminate the issues of finite precision arthmetic, as the example in my original post illustrates. Nor is binary any less accurate (in fact, it minimizes the relative inaccuracy). You just have to remember that approximation issues may creep into problems where you otherwise would not expect them.
>
> Jerry
>
> "Earl Kiosterud" wrote:
>
>> I've often wondered, aren't computers fast enough these days that they could do decimal
>> math, instead of having to do binary conversions? Even if the decimal math had do be
>> done
>> in software? Better yet, of course, would be for the CPU do be able to do it natively,
>> as
>> IBM systems were doing decades ago. It seems to me we mess around with

having to handle
the
errors for a far greater time than additional time the computer might need,
even for a
large
job. Just a thought.
−−
Earl Kiosterud
www.smokeylake.com

−−

"Jerry W. Lewis" <post_a_reply@xxxxxxxxxxxxx> wrote in message
news:BFE57CA3−7FF1−4C53−BC8E−EE0731DAB826@xxxxxxxxxxxxxxx

> This is a natural consequence of finite precision arithmetic.
> Excel is
> behaving exactly according to the IEEE standard, as most
> computers and
> software in the world do.
>
> If you had a calculator that only carried 4 decimal figures of
> accuracy,
> then you would only expect
> (1+x)−1−x
> to be zero if x could be represented in 3 decimal figures. For
> instance, if
> x=1/3, then you would have
> 1.333−1−0.3333
> which should not produce zero.
>
> The only potential surprise here is that 0.01 cannot be exactly
> represented.
> But computers work in binary where most terminating
> decimal fractions are
> non−terminating binary fractions. IEEE double precision
> uses 53 bits to
> approximate numbers, so the decimal equivalents of the
> binary approximations
> to 1.01 and 0.01 are
> 1.01000000000000000888178419700125232338905334472656250
> 0.01000000000000000020816681711721685132943093776702880859375
> so that your calculation correctly results in
> 8.673617379884035472059622406959533691406250E−18
> which Excel reports to its documented display limit of 15
> digits as
> 8.67361737988404E−18
>
> You get 0 from =1.01−1−0.01 (removing the parentheses)
> because Excel is
> trying to be "helpful" and assumes that a final difference
> between 2 numbers

that are equal to 15 digits is not meaningful. With the parentheses, the
parentheses rather than the subtraction is the final operation, so the fuzz
factor does not kick in to avoid introducing possibly unwanted inaccuracies
into the math.

Jerry

"baobob@xxxxxxxxxxx" wrote:

> Enter:
>
> =(1.01−1−0.01)
>
> To quote Paul Lynde, "I don't know what
> you got, but I got a sports
> shirt."
>
> ***

.