
Subject: Re: fix(4.70*100) is... 469

Posted by [James Kuyper](#) on Fri, 20 Apr 2007 16:26:07 GMT

[View Forum Message](#) <> [Reply to Message](#)

Qing wrote:

...

> (1) It is not a problem about just the FIX function:

>

> IDL> print, floor(4.7*100)

> 469

>

> (2) Can we always use ROUND instead of FIX or FLOOR? Then why the hell

> to have FIX and FLOOR to get confused?

No, you cannot always use ROUND instead of FIX or FLOOR, because, believe it or not, in some contexts the different result returned by FLOOR is more appropriate for what needs to be done. When I use FLOOR(x), I want any value that is greater than or equal to 469, and less than 470, to result in a return value of 469. The expression 4.7*100 has a value (at least on this hardware) which is less than 470. The number 470 can generally be represented exactly, but 4.7 cannot, and 4.7*100 is therefore extremely unlikely to calculate a value of exactly 470.

I use FLOOR far more often than ROUND, because most of the contexts where I might otherwise want to use ROUND are already handled correctly inside the formatted IO routines.

> (3) A precision issue? definitely need to read again at

> http://www.dfanning.com/math_tips/sky_is_falling.html .

> "There's nothing worse than trying to debug code and discovering weird

> results are related to the precision of the representation" - it can

> also be fun!

> But what about:

>

> IDL> print, fix(100D*4.7) ===== it can still be argued

> as a precision issue as long as you use 4.7 as an example !!!

> 469

It is an inherent consequence of the finite precision of the binary floating point representation that it cannot represent 4.7 exactly; the best it can do is a number either slightly larger or slightly smaller than 4.7. If your code depends upon that difference, you shouldn't be using floating point representations; use fixed point representations instead.

> (4) "... maybe apart from an insidious compiler bug, but that would

> never happen with IDL!"

- > does the problem happen in just IDL (on Windows, Linux, MacOS, ...)?
- > Do we have the same problem in C/C++, FORTRAN or even BASIC?

Yes, this happens in every computer language that allows decimal fractions to be stored in binary format.

- > Lets continue the hunt... it may not be just fun. Is it possible that
- > the difference between 469 and 470 could end up sending a satellite
- > off its track :-((

Only when code is written which handles incorrectly the characteristics of floating point data. Which, admittedly, is not rare.
