Subject: Re: Numbers from nowhere?
Posted by bennette on Fri, 15 Feb 2008 21:12:02 GMT
View Forum Message <> Reply to Message

On Feb 15, 10:08 am, Conor <cmanc...@gmail.com> wrote:
> I'm just curious:
>
> print,3.3,format='(f30.28)'
> 3.2999999523162841796875000000
>
> Obviously float is only good to ~6 decimal places, but IDL will still
> print out additional digits. It seems that it does reach a point
> where IDL will only print out zeroes. I'm just curious - where do
> these extra digits come from? Does it come from rounding errors
> converting binary to decimal? Or does it come from some other
> nefarious location?

I think that you hit the nail on the head. There is a nice article on
decimal to binary conversion on David Fanning's sight.
http://www.dfanning.com/math_tips/sky_is_falling.html

Eric

Eric Bennett
National Institutes of Health
bennette at nih.nhlbi.gov

Subject: Re: Numbers from nowhere?
Posted by Paul Van Delst[1] on Fri, 15 Feb 2008 21:31:30 GMT
View Forum Message <> Reply to Message

Conor wrote:
> I'm just curious:
>
> print,3.3,format='(f30.28)'
> 3.2999999523162841796875000000
>
> Obviously float is only good to ~6 decimal places, but IDL will still
> print out additional digits. It seems that it does reach a point
> where IDL will only print out zeroes. I'm just curious - where do
> these extra digits come from? Does it come from rounding errors
> converting binary to decimal? Or does it come from some other
> nefarious location?

Cosmic rays randomly flipping the bits beyond the stated precision?

:o)

Subject: Re: Numbers from nowhere?
Posted by David Fanning on Fri, 15 Feb 2008 22:51:16 GMT
View Forum Message <> Reply to Message

Paul van Delst writes:

> Cosmic rays randomly flipping the bits beyond the stated precision?

Teams of coyotes, given jobs like this to satisfy
their playful nature, but still keep them off the
street, more likely.

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming (www.dfanning.com)
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Numbers from nowhere?
Posted by elwood on Sun, 17 Feb 2008 20:19:44 GMT
View Forum Message <> Reply to Message

I'm glad that the Coyotes have something useful to do, besides keeping
me awake
all night in the Anza Borrego desert (it was wonderful to hear them
all night actually!)

But my question is more pointed:
if you assign x=3.3
and you know apriori that the floating point data type
will not have enough bits to store this number precisely,
why does "print" show this number as 3.3?

Is it because it is rounding off with some kind of algorithm?
Or am I misinterpreting how this number is being stored?

Curious,
Elisha

On Feb 15, 4:51 pm, David Fanning <n...@dfanning.com> wrote:

> Paul van Delst writes:
>> Cosmic rays randomly flipping the bits beyond the stated precision?
>
> Teams of coyotes, given jobs like this to satisfy
> their playful nature, but still keep them off the
> street, more likely.
>
> Cheers,
>
> David
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming (www.dfanning.com)
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Numbers from nowhere?
Posted by David Fanning on Sun, 17 Feb 2008 21:53:56 GMT
View Forum Message <> Reply to Message

elwood writes:

> But my question is more pointed:
> if you assign x=3.3
> and you know apriori that the floating point data type
> will not have enough bits to store this number precisely,
> why does "print" show this number as 3.3?
>
> Is it because it is rounding off with some kind of algorithm?
> Or am I misinterpreting how this number is being stored?

I presume it is because whatever number *is* stored,
when rounded to the 7-8 significant figures a float
can accurately represent, comes out to 3.300000.

```
IDL> x=3.3
IDL> print, x, format='(f0)'
   3.300000

IDL> print, x, format='(f12.10)'
   3.2999999523
```

Cheers,

David
--
David Fanning, Ph.D.

Subject: Re: Numbers from nowhere?
Posted by Sven Utcke on Thu, 21 Feb 2008 13:06:48 GMT
View Forum Message <> Reply to Message

David Fanning <news@dfanning.com> writes:

> elwood writes:
>
>> But my question is more pointed: if you assign x=3.3 and you know
>> apriori that the floating point data type will not have enough bits
>> to store this number precisely, why does "print" show this number
>> as 3.3?
>
> I presume it is because whatever number *is* stored, when rounded to
> the 7-8 significant figures a float can accurately represent, comes
> out to 3.300000.

What number _is_ stored, actually? Assuming we are talking ieee, we
have one bit for the sign, 8 for the exponent, and 23 for the
mantissa. So what is 3.3?

3 = 11 = 1.1 * 2^1
0.3 = 0.0100110011001100110011001100110011001100...
which we see from

0.3*2 = _0_.6
0.6*2 = _1_.2
0.2*2 = _0_.4
0.4*2 = _0_.8
0.8*2 = _1_.6
0.6*2 = ...

so we get, combined,

3.3 = 1.10100110011001100110011 * 2^1

or

S | Exp + 127 | Mantissa without leading 1
0 | 1000000 | 1010011 00110011 00110011

which, if we recombine it, turns out to be 3.299999952316284179687 5

We can actually see this in IDL too:

IDL> print, byte(3.3,0,4)
  51 51 83 64
Which, if we rewrite it appropriately, turns out to be:

01000000 01010011 00110011 00110011

which, recombined differently, is the above number :-)

Sven
--
  ___ _ _____ ___   Dr.-Ing. Sven Utcke
 / __| |/ / __| __| phone: +49 40 8998-5317
| (_ | ' <__ __ \ fax : +49 40 8994-5317 (NEW)
 ___|_|____|___/ http://www.desy.de/~utcke   (to come)

Subject: Re: Numbers from nowhere?
Posted by elwood on Mon, 25 Feb 2008 17:57:14 GMT
View Forum Message <> Reply to Message

Thanks,
very succinct and clear explanation.
Fun with Binary and Computer Precision is what I summarize
the topic as ;-)

-Elisha

On Feb 21, 7:06 am, Sven Utcke <utcke+n...@informatik.uni-hamburg.de>
wrote:
> David Fanning <n...@dfanning.com> writes:
>> elwood writes:
>
>>> But my question is more pointed: if you assign x=3.3 and you know
>>> apriori that the floating point data type will not have enough bits
>>> to store this number precisely, why does "print" show this number
>>> as 3.3?
>
>> I presume it is because whatever number *is* stored, when rounded to
>> the 7-8 significant figures a float can accurately represent, comes
>> out to 3.300000.
>
> What number _is_ stored, actually? Assuming we are talking ieee, we
> have one bit for the sign, 8 for the exponent, and 23 for the
> mantissa. So what is 3.3?
>
> 3 = 11 = 1.1 * 2^1

> 0.3 = 0.0100110011001100110011001100110011001100...
> which we see from
>
> 0.3*2 = _0_.6
> 0.6*2 = _1_.2
> 0.2*2 = _0_.4
> 0.4*2 = _0_.8
> 0.8*2 = _1_.6
> 0.6*2 = ...
>
> so we get, combined,
>
> 3.3 = 1.10100110011001100110011 * 2^1
>
> or
>
> S | Exp + 127 | Mantissa without leading 1
> 0 | 1000000 | 1010011 00110011 00110011
>
> which, if we recombine it, turns out to be 3.2999999523162841796875
>
> We can actually see this in IDL too:
>
> IDL> print, byte(3.3,0,4)
> 51 51 83 64
> Which, if we rewrite it appropriately, turns out to be:
>
> 01000000 01010011 00110011 T
> which, recombined differently, is the above number :-)
>
> Sven
> --
> ___ _ _____ ___ Dr.-Ing. Sven Utcke ___ ___ _____ __