
Subject: Re: strange behaviour of bytscl by large arrays

Posted by [lecacheux.alain](#) on Tue, 24 Apr 2012 14:50:46 GMT

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

On 23 avr, 22:22, Chris Torrence <gorth...@gmail.com> wrote:

> On Monday, April 23, 2012 10:14:21 AM UTC-6, fawltyl...@gmail.com wrote:

>

>> I think IDL's FINDGEN() implementation is wrong: it uses a float counter instead of an integer one. The following test shows the difference:

>

>> pro test

>> cpu, tpool_nthreads=1

>> n=10l^8

>> nn=n-1

>> a1=findgen(n) ; real FINDGEN()

>> a2=fltarr(n)

>> count=0.0

>> for j=0l, nn do a2[j]=count++ ; IDL's implementation

>> a3=fltarr(n)

>> count=0ll

>> for j=0l, nn do a3[j]=count++ ; better implementation

>> print, a1[nn], a2[nn], a3[nn], format='(3F15.3)'

>> end

>

>> (Multithreading must be disabled because the starting values for the threads are calculated as an integer. So the result of FINDGEN() depends on the number of your CPU cores, too :-)

>

>> regards,

>> Lajos

>

> Well, wrong is perhaps too strong of a word. The real word is "fast". I just did a test where I changed the internal implementation of FINDGEN to use an integer counter. The "float" counter is 4 times faster than using an integer counter and converting it to floats.

>

> However, perhaps we could look at the size of the input array, and switch to using the slower integer counter if it was absolutely necessary. I'll give it a thought.

>

> Thanks for reporting this!

>

> Cheers,

> Chris

> Exelis VIS

>

>

It is risky to write a statement like "findgen(n)" while n is larger than the inverse of the floating point precision (given in IDL by long(1/machar().eps)). This is true in any programming language. It is

mathematically incorrect to assume that such a "findgen" will behave as a "lindgen".

IDL is not "wrong" here, but rather clever. Isn't it ?
alx.
