
Subject: Inaccuracies

Posted by [Andy Loughe](#) on Mon, 13 Nov 1995 08:00:00 GMT

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

Ok, I am sure this has been discussed before, but let me start this thread again. I wish to create a 15-element vector which contains the numbers -1.4 to 1.4 by an increment of 0.2 I also wish the sum of these elements to be zero (No, this isn't the new math). Here is what I tried...

TRIAL #1

=====

```
IDL> a = findgen(15)*.2 - 1.4
```

```
IDL> print, total(a)
```

```
7.15256e-07
```

Hmmm! Not so good.

TRIAL #2

=====

```
IDL> a = dindgen(15)*(.2D)-1.4D
```

```
IDL> print, total(a, /double)
```

```
4.4408921e-15
```

Ok, this is better but not correct.

And what are the values of a?

```
IDL> print, a
```

```
-1.4000000  -1.2000000  -1.0000000  -0.8000000  
-0.6000000  -0.4000000  -0.2000000  2.2204460e-16  
0.2000000   0.4000000   0.6000000   0.8000000  
1.0000000   1.2000000   1.4000000
```

I seem to have lost a zero somewhere, and for me this matters!!!

TRIAL #3

=====

What if I only needed 13 numbers between -1.2 and 1.2.

```
IDL> a = findgen(13)*.2 - 1.2
```

```
IDL> print, total(a)
```

```
0.000000
```

Now how can I get this to work for 15 numbers?

Maybe I am missing something here, but this kind of behavior makes IDL a bit problematical for scientific use. With only 15 numbers and double precision arithmetic, I can't believe this would fail in FORTRAN or C!

--

Andrew F. Loughe (afl@cdc.noaa.gov)
University of Colorado, CIRES * Campus Box 449 * Boulder, CO 80309
phone: (303) 492-0707 fax: (303) 497-7013

Subject: Re: Inaccuracies

Posted by [bowman](#) on Mon, 13 Nov 1995 08:00:00 GMT

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

In article <30A7BC4D.7018@cdc.noaa.gov>, Andy Loughe <afl@cdc.noaa.gov> wrote:

```
> IDL> a = findgen(15)*.2 - 1.4
> IDL> print, total(a)
> 7.15256e-07
```

~7 significant figures is what you expect from single precision arithmetic.

```
> IDL> a = dindgen(15)*(.2D)-1.4D
> IDL> print, total(a, /double)
> 4.4408921e-15
```

~15 is what you expect from double precision.

You can do this:

```
IDL> i = LINDGEN(15)*2L - 14L
IDL> print, i
   -14   -12   -10    -8    -6    -4
    -2     0     2     4     6     8
    10    12    14
IDL> c = DOUBLE(i)/DOUBLE(10)
IDL> print, c
-1.40000000 -1.20000000 -1.00000000 -0.80000000
-0.60000000 -0.40000000 -0.20000000  0.00000000
 0.20000000  0.40000000  0.60000000  0.80000000
 1.00000000  1.20000000  1.40000000
```

which gives you an exact 0 (division of 0 by anything should(!) be exactly 0), but the other terms are not necessarily exact, and you still get

```
IDL> print, total(c)
2.2204460e-16
```

which is the best you can hope for.

That's floating point arithmetic ...

Regards, Ken Bowman

--

Kenneth P. Bowman, Assoc. Prof.
Department of Meteorology
Texas A&M University
College Station, TX 77843-3150

409-862-4060
409-862-4132 fax
bowman@csrp.tamu.edu
