Subject: Re: findgen anomaly?
Posted by David Fanning on Fri, 10 Jul 2009 21:52:21 GMT
View Forum Message <> Reply to Message

## Chris Chronopoulos writes:

```
> this morning i decided to write up a short function that would return
> a list of values given a min, max and increment (like Mathematica's
> Range[] function) but it was giving me some problems. i narrowed it
> down to an oddity in findgen, which can described by the following
 example:
> IDL> min=5.5
> IDL> max=6.2
> IDL> increment=0.1
> IDL> print, (max-min)/increment+1
      8.00000
  IDL> print,findgen(8.00000)
      0.00000
                 1.00000
                             2.00000
                                         3.00000
                                                     4.00000
>
  5.00000
      6.00000
                  7.00000
  IDL> print, findgen((max-min)/increment+1)
                  1.00000
                             2.00000
                                         3.00000
      0.00000
                                                     4.00000
  5.00000
      6.00000
>
>
> now, by any reasonable system of logic, shouldn't the last two lines
  produce the same result? the correct result, of course, is the first
> on that goes all the way up to 7.00000, the weird thing is that if you
> change max from 6.2 to 6.3, it works fine. it seems there are certain
> values that give it problems, while others work fine.
>
> what is going on here?
```

Golly, it has been several weeks since we had to pull this article out:

http://www.dfanning.com/math\_tips/sky\_is\_falling.html

Cheers,

David

David Fanning, Ph.D. Fanning Software Consulting, Inc. Subject: Re: findgen anomaly?

```
Posted by Chris Chronopoulos on Fri, 10 Jul 2009 22:11:13 GMT
View Forum Message <> Reply to Message
On Jul 10, 2:52 pm, David Fanning <n...@dfanning.com> wrote:
> Chris Chronopoulos writes:
>> this morning i decided to write up a short function that would return
>> a list of values given a min, max and increment (like Mathematica's
>> Range[] function) but it was giving me some problems. i narrowed it
>> down to an oddity in findgen, which can described by the following
>> example:
>> IDL> min=5.5
>> IDL> max=6.2
>> IDL> increment=0.1
>> IDL> print, (max-min)/increment+1
       8.00000
>>
>> IDL> print, findgen(8.00000)
                   1.00000
       0.00000
                              2.00000
                                                      4.00000
                                          3.00000
>> 5.00000
       6.00000
                   7.00000
>> IDL> print, findgen((max-min)/increment+1)
       0.00000
                   1.00000
                              2.00000
                                          3.00000
                                                      4.00000
>>
>> 5.00000
       6.00000
>>
>> now, by any reasonable system of logic, shouldn't the last two lines
>> produce the same result? the correct result, of course, is the first
>> on that goes all the way up to 7.00000. the weird thing is that if you
>> change max from 6.2 to 6.3, it works fine. it seems there are certain
>> values that give it problems, while others work fine.
>
>> what is going on here?
>
 Golly, it has been several weeks since we had to pull
  this article out:
>
>
   http://www.dfanning.com/math_tips/sky_is_falling.html
>
>
 Cheers,
>
> David
```

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

Thanks, David. That makes sense - I guess FINDGEN floors its input instead of rounding it. I stuck a ROUND in there, and it works like I want it to.

Subject: Re: findgen anomaly?
Posted by Kenneth P. Bowman on Sat, 11 Jul 2009 12:07:23 GMT
View Forum Message <> Reply to Message

## In article

<70fcb4fd-047f-43d5-a67a-2e2a5ccefa62@h8g2000yqm.googlegroups.com>, Chris Chronopoulos <chronopoulos.chris@gmail.com> wrote:

- > Thanks, David. That makes sense I guess FINDGEN floors its input
- > instead of rounding it. I stuck a ROUND in there, and it works like I
- > want it to.

Strictly speaking, I think it LONGs the argument (rounds toward zero by truncating the fractional part). Because a negative argument would generate an error, the difference is moot.

Ken Bowman