Subject: NaN problems
Posted by David Lopez Pons on Thu, 21 Apr 2005 13:37:44 GMT
View Forum Message <> Reply to Message

Hi everybody.
I have a problem detecting NaN values. When I write:

```
err=findgen(10)-5
res=err^2.5
print,res

   NaN       NaN       NaN       NaN       NaN
0.00000   1.00000   5.65685    15.5885   32.0000
```

Until here everything is OK, but now i want to look for Nan values in my result so I write:

```
print,where(res EQ !values.f_nan)
       -1
```

I can see the NaN values in res. Why they are not equal to !values.f_nan?

Subject: Re: NaN problems
Posted by David Fanning on Sat, 23 Apr 2005 03:38:31 GMT
View Forum Message <> Reply to Message

Y.T. writes:

> I don't have access to IDL on a win-box, but I have a (potentially
> stupid) question: On that page of yours where you write
>
> IDL> print, array ne array
>   0   0   0   0   0
>
> have you tried the converse, i.e.
>
> IDL> print, array eq array
>
> ?
>
> I am asking since, while it is true that a NaN is not equal to a NaN it
> does not seem to follow (to my wrinkled brain) that a NaN is
> necessarily UNequal to a NaN.
>
> In your version you test for the latter, and I'm kinda curious what
> would happen if you tried the former...

Well, looks like some things have changed since that
article was written. Here is what I get in IDL 6.1 on Windows:

```
IDL> a = [ 1.0, 2.0, !Values.F_NAN, 4.0, !Values.F_NAN ]
IDL> print, a ne a
   0   0   1   0   1
 % Program caused arithmetic error: Floating illegal operand
IDL> print, a eq a
   1   1   0   1   0
% Program caused arithmetic error: Floating illegal operand
```

Consistent, anyway. :-)

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/

Subject: Re: NaN problems
Posted by R.Bauer on Sat, 23 Apr 2005 19:48:39 GMT
View Forum Message <> Reply to Message

David Fanning wrote:

> Y.T. writes:
>
>> I don't have access to IDL on a win-box, but I have a (potentially
>> stupid) question: On that page of yours where you write
>>
>> IDL> print, array ne array
>>   0   0   0   0   0
>>
>> have you tried the converse, i.e.
>>
>> IDL> print, array eq array
>>
>> ?
>>
>> I am asking since, while it is true that a NaN is not equal to a NaN it
>> does not seem to follow (to my wrinkled brain) that a NaN is
>> necessarily UNequal to a NaN.
>>
>> In your version you test for the latter, and I'm kinda curious what
>> would happen if you tried the former...

>
> Well, looks like some things have changed since that
> article was written. Here is what I get in IDL 6.1 on Windows:
>
> IDL> a = [1.0, 2.0, !Values.F_NAN, 4.0, !Values.F_NAN]
> IDL> print, a ne a
> 0 0 1 0 1
> % Program caused arithmetic error: Floating illegal operand
> IDL> print, a eq a
> 1 1 0 1 0
> % Program caused arithmetic error: Floating illegal operand
>
> Consistent, anyway. :-)
>
> Cheers,
>
> David

I got the same result without the warnings on the linux Version.
Probably we both should try it with the actual 6.1.1 version. For windows
it's a 2MB download and I don't know why it is for linux 60MB. ;-)

cheers

Reimar

--
Forschungszentrum Juelich
email: R.Bauer@fz-juelich.de
http://www.fz-juelich.de/icg/icg-i/
 ========== ======
a IDL library at ForschungsZentrum Juelich
 http://www.fz-juelich.de/icg/icg-i/idl_icglib/idl_lib_intro. html

---

Subject: Re: NaN problems
Posted by yp on Mon, 25 Apr 2005 23:49:44 GMT
View Forum Message <> Reply to Message

David Lopez Pons wrote:
> Hi everybody.
> I have a problem detecting NaN values. When I write:
>
> err=findgen(10)-5
> res=err^2.5
> print,res

>
>   NaN   NaN   NaN   NaN   NaN
> 0.00000   1.00000   5.65685   15.5885   32.0000
>
> Until here everything is OK, but now i want to look for Nan values in
my
> result so I write:
>
> print,where(res EQ !values.f_nan)
>   -1
>
> I can see the NaN values in res. Why they are not equal to
!values.f_nan?

Use
print,where(FINITE(res) EQ 0)

instead. FINITE returns 1 or 0 if the argument is True (Finite) or
False (NaN or Inf) respectively.
Alternatively you can use:
print,where(FINITE(res,/NAN) EQ 1)

yas

Subject: Re: NaN problems
Posted by yp on Mon, 25 Apr 2005 23:51:48 GMT
View Forum Message <> Reply to Message

David Lopez Pons wrote:
> Hi everybody.
> I have a problem detecting NaN values. When I write:
>
> err=findgen(10)-5
> res=err^2.5
> print,res
>
>   NaN   NaN   NaN   NaN   NaN
> 0.00000   1.00000   5.65685   15.5885   32.0000
>
> Until here everything is OK, but now i want to look for Nan values in
my
> result so I write:
>
> print,where(res EQ !values.f_nan)
>   -1
>
> I can see the NaN values in res. Why they are not equal to

!values.f_nan?

Use
print,where(FINITE(res) EQ 0)

instead. FINITE returns 1 or 0 if the argument is True (Finite) or
False (NaN or Inf) respectively.
Alternatively you can use:
print,where(FINITE(res,/NAN) EQ 1)

yas

Subject: Re: NAN Problem
Posted by jeanh on Wed, 13 Jan 2010 20:28:34 GMT
View Forum Message <> Reply to Message

David Fanning wrote:
> Folks,
>
> Consider this sequence of commands on my IDL 7.0.1
> LINUX version.
>
> IDL> a = Findgen(11)
> IDL> a[0] = !Values.F_NAN
> IDL> Print, Min(a), Max(a)
> NaN NaN
> IDL> b = Findgen(11)
> IDL> b[1] = !Values.F_NAN
> IDL> Print, Min(b), Max(b)
> 0.00000 10.00000
>
> What do you make of that? This is giving me a great
> deal of touble today while trying to elminate bad
> values from an image. :-(
>
> I have the same results with my Windows IDL 7.1.2
> version, except I do get a warning about a floating
> illegal operand with variable b, which I don't get
> on LINUX.
>
> Cheers,
>
> David

hi,

heu, hum.... the help for min and max says "Note

If the MIN function is run on an array containing NaN values and the NAN keyword is not set, an invalid result will occur."

Jean

Subject: Re: NAN Problem
Posted by David Fanning on Wed, 13 Jan 2010 20:39:28 GMT
View Forum Message <> Reply to Message

jeanh writes:

> heu, hum.... the help for min and max says "Note
> If the MIN function is run on an array containing NaN values and the NAN
> keyword is not set, an invalid result will occur."

I was sort of hoping the "invalid result" would make
itself known to me. :-(

Cheers,

David

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

Subject: Re: NAN Problem
Posted by penteado on Wed, 13 Jan 2010 20:59:11 GMT
View Forum Message <> Reply to Message

On Jan 13, 6:23 pm, David Fanning <n...@dfanning.com> wrote:
> Folks,
>
> Consider this sequence of commands on my IDL 7.0.1
> LINUX version.
>
> IDL> a = Findgen(11)
> IDL> a[0] = !Values.F_NAN
> IDL> Print, Min(a), Max(a)
> NaN NaN
> IDL> b = Findgen(11)
> IDL> b[1] = !Values.F_NAN

> IDL> Print, Min(b), Max(b)
> 0.00000 10.00000
>
> What do you make of that? This is giving me a great
> deal of touble today while trying to elminate bad
> values from an image. :-(
>
> I have the same results with my Windows IDL 7.1.2
> version, except I do get a warning about a floating
> illegal operand with variable b, which I don't get
> on LINUX.

Interesting. I found the same here, with 7.1.1 in Linux. I had never noticed this behavior before.

I do not know how the min and max functions were implemented, but this would be the result of one algorithms that could be used for it. Since NaN does not compare larger or smaller than anything, the following would produce the same result:

```
function min_test,a
res=a[0]
for i=1,n_elements(a)-1 do if a[i] lt res then res=a[i]
return,res
end
```

When the NaN is the first element, it will be set as the initial value for the minimum, and since nothing is smaller, it will be kept as the result. Conversely, if the first value is not NaN, no NaN value that comes after will be smaller than it, so an NaN would not replace the minimum.

Subject: Re: NAN Problem
Posted by Foldy Lajos on Wed, 13 Jan 2010 21:00:17 GMT
View Forum Message <> Reply to Message

On Wed, 13 Jan 2010, David Fanning wrote:

> Consider this sequence of commands on my IDL 7.0.1
> LINUX version.
>
> IDL> a = Findgen(11)
> IDL> a[0] = !Values.F_NAN
> IDL> Print, Min(a), Max(a)
> NaN NaN
> IDL> b = Findgen(11)
> IDL> b[1] = !Values.F_NAN

> IDL> Print, Min(b), Max(b)
> 0.00000 10.00000
>
> What do you make of that? This is giving me a great
> deal of touble today while trying to elminate bad
> values from an image. :-(
>

The first array element causes the difference. You can reproduce the results:

```
min=a[0]
max=a[0]
for j=1,10 do begin
    if a[j] lt min then min=a[j]
    if a[j] gt max then max=a[j]
endfor
```

lt and gt are not defined for NaNs, they always return 0:

```
x=0.0
y=!Values.F_NAN
help, x lt y, x eq y, x gt y

<Expression>    BYTE     =    0
<Expression>    BYTE     =    0
<Expression>    BYTE     =    0
```

regards,
lajos

Subject: Re: NAN Problem
Posted by R.G.Stockwell on Thu, 14 Jan 2010 00:00:19 GMT
View Forum Message <> Reply to Message

"David Fanning" <news@dfanning.com> wrote in message
news:MPG.25b7d9955ac199e398969e@news.giganews.com...
> Folks,
>
> Consider this sequence of commands on my IDL 7.0.1
> LINUX version.
>
> IDL> a = Findgen(11)
> IDL> a[0] = !Values.F_NAN
> IDL> Print, Min(a), Max(a)

> NaN NaN
> IDL> b = Findgen(11)
> IDL> b[1] = !Values.F_NAN
> IDL> Print, Min(b), Max(b)
> 0.00000 10.00000
>
> What do you make of that?

HOLY CRAP!!!

i swear i used to get nan as a the result of that.
I do get a :
% Program caused arithmetic error: Floating illegal operand

after that min(b) call. But now I am worried about how
invalid that result is (and how diligent i have been in using
nan keywords).

cheers,
bob

Subject: Re: NAN Problem
Posted by penteado on Thu, 14 Jan 2010 00:15:00 GMT
View Forum Message <> Reply to Message

On Jan 13, 10:00 pm, "R.G. Stockwell" <noem...@please.com> wrote:
> i swear i used to get nan as a the result of that.
> I do get a :
> % Program caused arithmetic error: Floating illegal operand
>
> after that min(b) call. But now I am worried about how
> invalid that result is (and how diligent i have been in using
> nan keywords).

Same here. It is probably one of the reasons I once had some
difficulty writing a reader/writer to a file format (ISIS cubes, in
case anyone is interested) which has several special values, that
could be translated to NaN.

Subject: Re: NAN Problem
Posted by R.Bauer on Thu, 14 Jan 2010 08:49:52 GMT
View Forum Message <> Reply to Message

David Fanning schrieb:

> Folks,
>
> Consider this sequence of commands on my IDL 7.0.1
> LINUX version.
>
> IDL> a = Findgen(11)
> IDL> a[0] = !Values.F_NAN
> IDL> Print, Min(a), Max(a)
> NaN NaN
> IDL> b = Findgen(11)
> IDL> b[1] = !Values.F_NAN
> IDL> Print, Min(b), Max(b)
> 0.00000 10.00000
>
> What do you make of that? This is giving me a great
> deal of touble today while trying to elminate bad
> values from an image. :-(
>
> I have the same results with my Windows IDL 7.1.2
> version, except I do get a warning about a floating
> illegal operand with variable b, which I don't get
> on LINUX.
>
> Cheers,
>
> David
>
>
>

Hi David

For some reason the nan keyword was introduced

NAN

Set this keyword to cause the routine to check for occurrences of the IEEE floating-point values NaN or Infinity in the input data. Elements with the value NaN or Infinity are treated as missing data. (See Special Floating-Point Values (Application Programming) for more information on IEEE floating-point values.)

```
IDL> print, min(a,/nan)
     1.00000
IDL> print, min(b,/nan)
     0.00000
```

I try to avoid nan values.

Because not all operators do have the possibility to set nan operations
by a keyword. Frank has added some
 http://www.fz-juelich.de/icg/icg-1/idl_icglib/idl_source/idl _work/fh_lib/f_ge.pro
functions because he likes nan.

cheers
Reimar
