Subject: Re: VALUE_LOCATE and NaNs
Posted by lecacheux.alain on Thu, 25 Oct 2012 09:34:04 GMT
View Forum Message <> Reply to Message

Le jeudi 25 octobre 2012 11:14:47 UTC+2, Fab a écrit :
> Dear IDLers,
>
>
>
> I am quite an adept of using NaNs as missing values for my data, because
>
> many IDL routines go along with them. However, some of the routines are
>
> not really documented about how they handle NaNs.
>
> See the following example:
>
>
>
> IDL> data = FINDGEN(10) & data[0] = !VALUES.F_NAN
>
> IDL> p = VALUE_LOCATE(FINDGEN(10), data) & print, p[0]
>
>   -1
>
> IDL> p = VALUE_LOCATE(INDGEN(10), data) & print, p[0]
>
>   0
>
> % Program caused arithmetic error: Floating illegal operand
>
>
>
> Which is quite dangerous! If I didn't debug my code to find the origin
>
> of the Floating illegal operand warning, value_locate's output would be
>
> wrong.
>
>
>
> Someone else than me thinking of this as a bug?
>
>
>
> Thanks,
>
>

>
> Fab

Please note that:

```
IDL> p = VALUE_LOCATE(LINDGEN(10), data) & print, p[0]
     -1
```

gives the correct result.

alx.

Subject: Re: VALUE_LOCATE and NaNs
Posted by Fabzi on Thu, 25 Oct 2012 09:57:30 GMT

View Forum Message <> Reply to Message

On 10/25/2012 11:34 AM, alx wrote:

```
 > Please note that:
 > IDL> p = VALUE_LOCATE(LINDGEN(10), data) & print, p[0]
 >      -1
 > gives the correct result.
 >
 > alx.
```

Well this is even more confusing. But in my case it doesn't:

```
IDL>  data = FINDGEN(10) &  data[0] = !VALUES.F_NAN
IDL> p = VALUE_LOCATE(LINDGEN(10), data) & print, p[0]
           -1
% Program caused arithmetic error: Floating illegal operand
IDL> print, !VERSION
{ x86_64 linux unix linux 7.1.1 Aug 21 2009      64      64}
```

Subject: Re: VALUE_LOCATE and NaNs
Posted by lecacheux.alain on Thu, 25 Oct 2012 11:10:15 GMT

View Forum Message <> Reply to Message

Le jeudi 25 octobre 2012 11:57:31 UTC+2, Fab a écrit :

```
>  On 10/25/2012 11:34 AM, alx wrote:
>
>>  Please note that:
>
>>  IDL> p = VALUE_LOCATE(LINDGEN(10), data) & print, p[0]
>
>>       -1
>
>>  gives the correct result.
```

>
>>
>
>> alx.
>
>
>
> Well this is even more confusing. But in my case it doesn't:
>
>
>
> IDL> data = FINDGEN(10) & data[0] = !VALUES.F_NAN
>
> IDL> p = VALUE_LOCATE(LINDGEN(10), data) & print, p[0]
>
>   -1
>
> % Program caused arithmetic error: Floating illegal operand
>
> IDL> print, !VERSION
>
> { x86_64 linux unix linux 7.1.1 Aug 21 2009   64   64}

Well, I checked with
{ x86_64 Win32 Windows Microsoft Windows 8.2.1 Aug 20 2012   64   64}
alx.

Subject: Re: VALUE_LOCATE and NaNs
Posted by Kai Muehlbauer on Thu, 25 Oct 2012 13:41:05 GMT
View Forum Message <> Reply to Message

Am 25.10.2012 13:10, schrieb alx:
> Le jeudi 25 octobre 2012 11:57:31 UTC+2, Fab a ï¿½crit :
>> On 10/25/2012 11:34 AM, alx wrote:
>>
>>> Please note that:
>>
>>> IDL> p = VALUE_LOCATE(LINDGEN(10), data) & print, p[0]
>>
>>>   -1
>>
>>> gives the correct result.
>>
>>>
>>
>>> alx.
>>

```
>>
>>
>>  Well this is even more confusing. But in my case it doesn't:
>>
>>
>>
>>  IDL>  data = FINDGEN(10) &  data[0] = !VALUES.F_NAN
>>
>>  IDL> p = VALUE_LOCATE(LINDGEN(10), data) & print, p[0]
>>
>>            -1
>>
>>  % Program caused arithmetic error: Floating illegal operand
>>
>>  IDL> print, !VERSION
>>
>>  { x86_64 linux unix linux 7.1.1 Aug 21 2009      64      64}
>
>  Well, I checked with
>  { x86_64 Win32 Windows Microsoft Windows 8.2.1 Aug 20 2012      64      64}
>  alx.
>
```

I can confirm:

```
IDL> print, !VERSION
{ x86_64 linux unix linux 7.1.1 Aug 21 2009      64      64}

IDL> data = FINDGEN(10) & data[0] = !VALUES.F_NAN
IDL> p = VALUE_LOCATE(LINDGEN(10), data) & print, p[0]
           -1
% Program caused arithmetic error: Floating illegal operand
IDL>
```

Cheers,
Kai

Subject: Re: VALUE_LOCATE and NaNs
Posted by Fabzi on Thu, 25 Oct 2012 13:48:04 GMT

View Forum Message <> Reply to Message

On 10/25/2012 03:41 PM, Kai Muehlbauer wrote:

```
>
>  I can confirm:
>
>  IDL> print, !VERSION
>  { x86_64 linux unix linux 7.1.1 Aug 21 2009      64      64}
```

```
>
> IDL> data = FINDGEN(10) & data[0] = !VALUES.F_NAN
> IDL> p = VALUE_LOCATE(LINDGEN(10), data) & print, p[0]
>           -1
> % Program caused arithmetic error: Floating illegal operand
> IDL>
>
> Cheers,
> Kai
>
```

At least the answer is right, but the warning is there ;-)
So three different behaviours of value locate with three input types:

```
IDL> data = FINDGEN(10) & data[0] = !VALUES.F_NAN
IDL>  p = VALUE_LOCATE(INDGEN(10), data) & print, p[0]
       0
% Program caused arithmetic error: Floating illegal operand
IDL>  p = VALUE_LOCATE(LINDGEN(10), data) & print, p[0]
          -1
% Program caused arithmetic error: Floating illegal operand
IDL>  p = VALUE_LOCATE(FINDGEN(10), data) & print, p[0]
          -1
```

Subject: Re: VALUE_LOCATE and NaNs
Posted by Craig Markwardt on Thu, 25 Oct 2012 14:30:22 GMT

View Forum Message <> Reply to Message

On Thursday, October 25, 2012 5:14:47 AM UTC-4, Fab wrote:

```
> Dear IDLers,
>
> I am quite an adept of using NaNs as missing values for my data, because
> many IDL routines go along with them. However, some of the routines are
> not really documented about how they handle NaNs.
>
> See the following example:
> IDL> data = FINDGEN(10) &  data[0] = !VALUES.F_NAN
> IDL> p = VALUE_LOCATE(FINDGEN(10), data) & print, p[0]
>           -1
> IDL> p = VALUE_LOCATE(INDGEN(10), data) & print, p[0]
>            0
> % Program caused arithmetic error: Floating illegal operand
>
>
>
> Which is quite dangerous! If I didn't debug my code to find the origin
> of the Floating illegal operand warning, value_locate's output would be
```

> wrong.
>
>
>
> Someone else than me thinking of this as a bug?

NAN is neither greater than or less than any other finite number. For the purposes of VALUE_LOCATE(), there is no way to indicate "complete failure" other than the message you saw. I think the results of VALUE_LOCATE() are undefined, but this should be documented more explicitly in the reference documentation.

The lesson is: you can't use NAN's with VALUE_LOCATE().

Craig

Subject: Re: VALUE_LOCATE and NaNs
Posted by lecacheux.alain on Thu, 25 Oct 2012 14:59:48 GMT

View Forum Message <> Reply to Message

Le jeudi 25 octobre 2012 16:30:22 UTC+2, Craig Markwardt a écrit :
> On Thursday, October 25, 2012 5:14:47 AM UTC-4, Fab wrote:
>
>> Dear IDLers,
>
>>
>
>> I am quite an adept of using NaNs as missing values for my data, because
>
>> many IDL routines go along with them. However, some of the routines are
>
>> not really documented about how they handle NaNs.
>
>>
>
>> See the following example:
>
>> IDL> data = FINDGEN(10) & data[0] = !VALUES.F_NAN
>
>> IDL> p = VALUE_LOCATE(FINDGEN(10), data) & print, p[0]
>
>> -1
>
>> IDL> p = VALUE_LOCATE(INDGEN(10), data) & print, p[0]
>
>> 0
>
>> % Program caused arithmetic error: Floating illegal operand

>
>>
>
>>
>
>>
>
>> Which is quite dangerous! If I didn't debug my code to find the origin
>
>> of the Floating illegal operand warning, value_locate's output would be
>
>> wrong.
>
>>
>
>>
>
>>
>
>> Someone else than me thinking of this as a bug?
>
>
>
> NAN is neither greater than or less than any other finite number. For the purposes of VALUE_LOCATE(), there is no way to indicate "complete failure" other than the message you saw. I think the results of VALUE_LOCATE() are undefined, but this should be documented more explicitly in the reference documentation.
>
>
>
> The lesson is: you can't use NAN's with VALUE_LOCATE().
>
>
>
> Craig

In addition, there is one more inconsistency: 'value_locate' is considering NAN as the smallest value, while 'sort' is doing the opposite. Both being wrong as Craig said.
In IDL 8.2.1 and Win7, no warning/error messages are provided.

```
IDL> data=findgen(10) & data[1] = !Values.F_NAN
IDL> print,value_locate(findgen(10),data)
           0          -1           2           3           4           5           6           7           8           9
IDL> print,sort(data)
           0           2           3           4           5           6           7           8           9           1
```

alx.

Subject: Re: VALUE_LOCATE and NaNs

Posted by wlandsman on Thu, 25 Oct 2012 15:01:19 GMT

View Forum Message <> Reply to Message

On Thursday, October 25, 2012 10:30:22 AM UTC-4, Craig Markwardt wrote:

```
>
>  The lesson is: you can't use NAN's with VALUE_LOCATE().
>
```

Indeed the VALUE_LOCATE documentation says about the input vector

"A vector of monotonically increasing or decreasing values. Vector may be of type string, or any numeric type except complex, and may not contain the value NaN (not-a-number)"

--Wayne

Subject: Re: VALUE_LOCATE and NaNs

Posted by Jeremy Bailin on Thu, 25 Oct 2012 15:05:13 GMT

View Forum Message <> Reply to Message

On 10/25/12 9:48 AM, Fab wrote:

```
>  On 10/25/2012 03:41 PM, Kai Muehlbauer wrote:
>>
>>  I can confirm:
>>
>>  IDL> print, !VERSION
>>  { x86_64 linux unix linux 7.1.1 Aug 21 2009      64      64}
>>
>>  IDL> data = FINDGEN(10) & data[0] = !VALUES.F_NAN
>>  IDL> p = VALUE_LOCATE(LINDGEN(10), data) & print, p[0]
>>            -1
>>  % Program caused arithmetic error: Floating illegal operand
>>  IDL>
>>
>>  Cheers,
>>  Kai
>>
>
>  At least the answer is right, but the warning is there ;-)
>  So three different behaviours of value locate with three input types:
>
>  IDL> data = FINDGEN(10) & data[0] = !VALUES.F_NAN
>  IDL>  p = VALUE_LOCATE(INDGEN(10), data) & print, p[0]
>            0
>  % Program caused arithmetic error: Floating illegal operand
>  IDL>  p = VALUE_LOCATE(LINDGEN(10), data) & print, p[0]
```

>   -1
> % Program caused arithmetic error: Floating illegal operand
> IDL> p = VALUE_LOCATE(FINDGEN(10), data) & print, p[0]
>   -1
>
>

I agree with Craig - there is no well-defined answer for what VALUE_LOCATE should return when faced with a NaN. But it also shouldn't give different answers depending on the type of location array.

-Jeremy.

Subject: Re: VALUE_LOCATE and NaNs
Posted by Fabzi on Thu, 25 Oct 2012 15:07:22 GMT

View Forum Message <> Reply to Message

On 10/25/2012 05:01 PM, wlandsman wrote:
> Indeed the VALUE_LOCATE documentation says about the input vector
>
>
> "A vector of monotonically increasing or decreasing values. Vector
> may be of type string, or any numeric type except complex, and may
> not contain the value NaN (not-a-number)"
>
> --Wayne

Yes, but the Value array contains NaNs, not the vector. Anyway, I got the lesson and I will check for NaNs before using value_locate. Thanks!