
Subject: REFORM: new subscripts must not change the number elements in array
Posted by [g.nacarts](#) on Mon, 02 Mar 2015 15:19:30 GMT

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

I have a 2D array of A=[200,200]

I am running REFORM but get the following error
"REFORM: New subscripts must not change the number elements in A"

This is what I am doing
A = reform(A, 100L, 200L*200L, /overwrite)

Why isn't it working?

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [Moritz Fischer](#) on Mon, 02 Mar 2015 15:26:19 GMT

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

Am 02.03.2015 um 16:19 schrieb g.nacarts@gmail.com:

> I have a 2D array of A=[200,200]
>
> I am running REFORM but get the following error
> "REFORM: New subscripts must not change the number elements in A"
>
> This is what I am doing
> A = reform(A, 100L, 200L*200L, /overwrite)
>
> Why isn't it working?
>

You are trying to reform an 40,000 element array to an array with
 $100*200*200 = 4,000,000$ elements.

You could for example do

A = reform(A, 100L, 400L, /overwrite)

as $200*200 = 100*400$

REFORM just changes the way IDL looks at an array.

To interpolate/blow up your array look at rebin!

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [Helder Marchetto](#) on Mon, 02 Mar 2015 15:31:08 GMT

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

On Monday, March 2, 2015 at 4:19:32 PM UTC+1, g.na...@gmail.com wrote:

> I have a 2D array of A=[200,200]
>
> I am running REFORM but get the following error
> "REFORM: New subscripts must not change the number elements in A"
>
> This is what I am doing
> A = reform(A, 100L, 200L*200L, /overwrite)
>
> Why isn't it working?

Because 4000 is not 4000000.

Let me show:

```
IDL> A=randomu(s,200,200)
```

```
IDL> print, product(size(a, /dimensions), /preserve_type)  
40000
```

```
IDL> print, product([100L, 200L*200L], /preserve_type)  
4000000
```

If your array is has 200 x 200 elements, the total is 40000.

Now with reform you want to make it 100 x (200x200)... this is 4000000.

Maybe (maybe!) what you want is:

```
a = reform(a, 100L, 200L*2L, /overwrite)
```

Cheers,
Helder

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [g.nacarts](#) on Mon, 02 Mar 2015 15:43:16 GMT

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

There is one thing I don't understand. Both of you suggest the same thing:

```
a = reform(a, 100L, 200L*2L, /overwrite)
```

You typed 200L*2L this is because we want 200x200 that's why you multiplied it by 2?

What I want it's 100 x (200x200). So 100 2D images of dimensions [200,200]

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [g.nacarts](#) on Mon, 02 Mar 2015 15:46:08 GMT

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

Even with this

```
> a = reform(a, 100L, 200L*2L, /overwrite)
```

I still got the same error

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [Matthew Argall](#) on Mon, 02 Mar 2015 15:49:25 GMT

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

Another possibility is that you want to reform your array to be 400x1, then copy it 100 times, so that you have a 100x400. In this case, you want

```
IDL> A = randomu(s, 200, 200)
```

```
IDL> A = rebin(reform(A, 1, 400), 100, 400)
```

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [Helder Marchetto](#) on Mon, 02 Mar 2015 15:51:44 GMT

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

On Monday, March 2, 2015 at 4:46:09 PM UTC+1, g.na...@gmail.com wrote:

> Even with this

>> a = reform(a, 100l, 200l*2l, /overwrite)

>

> I still got the same error

What do you get if you type:

print, size(a)

before the reform command?

Helder

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [g.nacarts](#) on Mon, 02 Mar 2015 16:14:58 GMT

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

> What do you get if you type:

> print, size(a)

> before the reform command?

>

> Helder

I got

```
2      200      200      4      40000
```

Subject: Re: REFORM: new subscripts must not change the number elements in

array

Posted by [Helder Marchetto](#) on Mon, 02 Mar 2015 16:19:46 GMT

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

On Monday, March 2, 2015 at 5:14:59 PM UTC+1, g.na...@gmail.com wrote:

```
>> What do you get if you type:
>> print, size(a)
>> before the reform command?
>>
>> Helder
>
> I got
>
> 2      200      200      4      40000
```

I don't see how you can get an error with the `reform()` command:

```
IDL> a = randomu(s, 200,200)
IDL> print, size(a)
      2      200      200      4      40000
IDL> help, reform(a, 100I, 200I*2I)
<Expression>  FLOAT    = Array[100, 400]
```

Cheers,
Helder

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [Helder Marchetto](#) on Mon, 02 Mar 2015 16:24:58 GMT

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

On Monday, March 2, 2015 at 4:43:17 PM UTC+1, g.na...@gmail.com wrote:

```
> There is one thing I don't understand. Both of you suggest the same thing:
> a = reform(a, 100I, 200I*2I, /overwrite)
> You typed 200I*2I this is because we want 200x200 that's why you multiplied it by 2?
> What I want it's 100 x (200x200). So 100 2D images of dimensions [200,200]
```

Ok, so before, I didn't see this message.

You're sort of changing things around here.

You started from ONE array with 200x200 elements and wanted to make something of 100x(40000) elements. This of course will never work.

What you really want is to make a 100x200x200 array from 100 arrays of 200x200. Correct?

But where do you get the 100 arrays of 200x200?

Try this:

```
finalArray = fltarr(100,200,200)
for i=0, 99 do begin
    ;generate the array
    a = randomu(s,200,200)
    ;put it in the final array
    finalArray[i,0,0] = reform(a,1,200,200)
endfor
```

Regards,
Helder

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [g.nacarts](#) on Mon, 02 Mar 2015 16:46:49 GMT

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

I didn't understand the logic of this `reform(a, 100I, 200I*2I)`.

If instead of `A=[200,200]` we have `A=[216,216]` and we want `120x(216x216)` then this is written as `reform(a, 120I, 216I*2I)`. Is that correct?

I am trying to understand how you end up with `reform(a, 100I, 200I*2I)`. Because when I typed this
`a = randomu(s, 216,216)`

`help, reform(a, 120I, 216I*2I,/overwrite)`

I got the same error "REFORM: New subscripts must not change the number elements in A."

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [Helder Marchetto](#) on Mon, 02 Mar 2015 17:01:51 GMT

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

On Monday, March 2, 2015 at 5:46:52 PM UTC+1, g.na...@gmail.com wrote:

> I didn't understand the logic of this `reform(a, 100I, 200I*2I)`.

> If instead of `A=[200,200]` we have `A=[216,216]` and we want `120x(216x216)` then this is written as `reform(a, 120I, 216I*2I)`. Is that correct?

>

> I am trying to understand how you end up with `reform(a, 100I, 200I*2I)`. Because when I typed this

> `a = randomu(s, 216,216)`

> `help, reform(a, 120I, 216I*2I,/overwrite)`

> I got the same error "REFORM: New subscripts must not change the number elements in A."

Look, it's getting difficult to help if the cards keep on changing.

Did you see my last message? You can do the same thing without using `reform()`, because I think you have not read the documentation of the `reform` command.

```
xSize = 216
ySize = 216
nImages = 120
finalArray = fltarr(xSize,ySize,nImages)
for i=0, nImages-1 do begin
    ;generate the array
    a = randomu(s,xSize,ySize)
    ;put it in the final array
    finalArray[0,0,i] = a
endfor
```

I didn't use the reform. For your information this:

```
finalArray[0,0,i] = a
is the same as this:
finalArray[*,* ,i] = a
Just a bit faster.
```

As an example, try this:

```
test = fltarr(3,3,5)
print, test
ones = fltarr(3,3)+1.0
test[0,0,2] = ones
print, test
```

Test contains 5 "images" of 3x3 pixels. When you first print it, you will see 5 images with zeros in each pixel. After
test[0,0,2] = ones
you will find that the third image is filled with ones.

I think this is described here:

http://www.exelisvis.com/docs/Array_Subscript_Ranges.html
in the section "Avoid using range subscripts for assignment".

Cheers,
Helder

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [David Fanning](#) on Mon, 02 Mar 2015 17:08:25 GMT

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

Helder writes:

> Look, it's getting difficult to help if the cards keep on changing.

Helder, I just wanted to let you know how much I appreciate your yeoman's work on this newsgroup. :-)

Cheers,

David

P.S. Did I mention how much I love skiing?

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.idlcoyote.com/>

Sepore ma de ni thue. ("Perhaps thou speakest truth.")

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [g.nacarts](#) on Mon, 02 Mar 2015 17:12:30 GMT

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

Yes, I saw your last message but I am interested to understand how you end up before with the `a = reform(a, 100I, 200I*2I, /overwrite)`. I followed your steps but I changed the dimensions to see if I understand well.

```
A=randomu(s,216,216)
```

```
IDL> print, product(size(a, /dimensions), /preserve_type)
46656
```

```
IDL> print, product([120L, 216L*216L], /preserve_type)
5598720
```

Then I tried to write it like this: `a = reform(a, 120I, 216I*2I, /overwrite)` but I got the error that I mentioned before.

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [Paul Van Delst\[1\]](#) on Mon, 02 Mar 2015 17:20:56 GMT

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

On 03/02/15 11:46, g.nacarts@gmail.com wrote:

```
> I didn't understand the logic of this reform(a, 100I, 200I*2I). If
> instead of A=[200,200] we have A=[216,216] and we want 120x(216x216)
> then this is written as reform(a, 120I, 216I*2I). Is that correct?
```

No. That is not correct.

Read:

<http://www.exelisvis.com/docs/REFORM.html>

The very first line of the description (actually, the only line) says:

<quote>

The REFORM function changes the dimensions of an array without changing the total number of elements.

</quote>

If you start with A=[200,200] and you want B=[200,200,100] (i.e. 100 sets of 200x200 arrays) then REFORM is not the command to use.

I think you want REBIN:

<http://www.exelisvis.com/docs/REBIN.html>

```
IDL> a=lindgen(4,4)
```

```
IDL> print, a
```

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

Now create 10 "copies" of a, stacked into a 3-D array:

```
IDL> b=rebin(a,4,4,10)
```

```
IDL> print, b[*,*,0]
```

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

```
IDL> print, b[*,*,6]
```

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

>

> I am trying to understand how you end up with reform(a, 100l,
> 200l*2l). Because when I typed this a = randomu(s, 216,216) help,
> reform(a, 120l, 216l*2l,/overwrite) I got the same error "REFORM: New
> subscripts must not change the number elements in A."

>

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [Paul Van Delst\[1\]](#) on Mon, 02 Mar 2015 17:25:47 GMT

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

With apologies for replying to my own reply, but if you really want an array B=[100,200,200] from a starting A=[200,200] array, you just use TRANSPOSE (<http://www.exelisvis.com/docs/TRANSPOSE.html>) on the result from REBIN:

```
IDL> help, b
```

```
B          LONG    = Array[4, 4, 10]
```

```
IDL> b = transpose(b,[2,0,1])
```

```
IDL> help, b
```

```
B          LONG    = Array[10, 4, 4]
```

```
IDL> print, b[0,*,*]
```

```
0
```

```
1
```

```
2
```

```
3
```

```
4
```

```
5
```

```
6
```

```
7
```

```
8
```

```
9
```

```
10
```

```
11
```

```
12
```

```
13
```

```
14
```

```
15
```

```
IDL> print, b[6,*,*]
```

```
0
```

```
1
```

```
2
```

```
3
```

```
4
```

```
5
```

```
6
```

```
7
```

```
8
```

```
9
```

```
10
```

```
11
```

12
13
14
15

On 03/02/15 12:20, Paul van Delst wrote:

> On 03/02/15 11:46, g.nacarts@gmail.com wrote:

>> I didn't understand the logic of this reform(a, 100l, 200l*2l). If

>> instead of A=[200,200] we have A=[216,216] and we want 120x(216x216)

>> then this is written as reform(a, 120l, 216l*2l). Is that correct?

>

> No. That is not correct.

>

> Read:

> <http://www.exelisvis.com/docs/REFORM.html>

>

> The very first line of the description (actually, the only line) says:

> <quote>

> The REFORM function changes the dimensions of an array without changing

> the total number of elements.

> </quote>

>

> If you start with A=[200,200] and you want B=[200,200,100] (i.e. 100

> sets of 200x200 arrays) then REFORM is not the command to use.

>

> I think you want REBIN:

> <http://www.exelisvis.com/docs/REBIN.html>

>

> IDL> a=lindgen(4,4)

> IDL> print, a

```
>      0      1      2      3
>      4      5      6      7
>      8      9     10     11
>     12     13     14     15
```

>

> Now create 10 "copies" of a, stacked into a 3-D array:

>

> IDL> b=rebin(a,4,4,10)

>

> IDL> print, b[*,*,0]

```
>      0      1      2      3
>      4      5      6      7
>      8      9     10     11
>     12     13     14     15
```

> IDL> print, b[*,*,6]

```
>      0      1      2      3
>      4      5      6      7
>      8      9     10     11
>     12     13     14     15
>
>>
>> I am trying to understand how you end up with reform(a, 100I,
>> 200I*2I). Because when I typed this a = randomu(s, 216,216) help,
>> reform(a, 120I, 216I*2I,/overwrite) I got the same error "REFORM: New
>> subscripts must not change the number elements in A."
>>
```

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [g.nacarts](#) on Mon, 02 Mar 2015 17:31:55 GMT

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

I read the documentation, and I noticed that The REFORM function changes the dimensions of an array without changing the total number of elements. That's what I want but I didn't understand how the others end up with a = reform(a, 100I, 200I*2I, /overwrite).

That's the thing that I want to understand. I used a different example because I wanted to make sure that I understand but it seems that I didn't because I wrote it like this a = reform(a, 120I, 216I*2I, /overwrite). I put 120 in the place of 100 and then 216 in place of 200 because I assumed that the matrix is A=[216,216]. But I got the error I mentioned above.

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [g.nacarts](#) on Mon, 02 Mar 2015 19:25:22 GMT

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

I used the example provided by mo who said that $200*200=100*400$. When a 2D array has dimensions [216,216] then $216*216=120*388.8$

when I tried a=reform(a,120I,388.8I,/overwrite) I got the old error

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [Phillip Bitzer](#) on Mon, 02 Mar 2015 19:34:03 GMT

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

On Monday, March 2, 2015 at 11:31:56 AM UTC-6, g.na...@gmail.com wrote:

```
> I read the documentation, and I noticed that The REFORM function changes the dimensions of
an array without changing the total number of elements. That's what I want but I didn't understand
```

how the others end up with `a = reform(a, 100l, 200l*2l, /overwrite)`.

>

Then you really need to read it again.

Your original question asks why an array that is 200x200 can not be reformed into an array that 100x40000.

On Monday, March 2, 2015 at 1:25:24 PM UTC-6, g.na...@gmail.com wrote:

> I used the example provided by mo who said that $200*200=100*400$. When a 2D array has dimensions [216,216] then $216*216=120*388.8$

>

> when I tried `a=reform(a,120l,388.8L,/overwrite)` I got the old error

Uh, a dimension with 388.8 elements?

As someone was said around here before he "retired," no, no, you're doing it all wrong.

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [David Fanning](#) on Mon, 02 Mar 2015 19:55:23 GMT

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

g.nacarts@gmail.com writes:

> I used the example provided by mo who said that $200*200=100*400$. When a 2D array has dimensions [216,216] then $216*216=120*388.8$

>

> when I tried `a=reform(a,120l,388.8L,/overwrite)` I got the old error

Are you familiar with "counting" numbers? 1,2,3,...

Array dimensions can only be specified with counting numbers. 388.8 is real number, but not a counting number. You can pick any two counting numbers to multiply together, so long as the TOTAL (also a counting number) is the same counting number you get when you multiply the original dimensions of the array together. Another way of saying this is that when you reform (bend, shape, twist, etc.) an array, it can't end up with more or less array elements than when you started.

If you DO want to end up with some other number of elements, then you are going to have to REBIN your array, rather than REFORM it.

I say this in the most gentle way possible, but are you sure you are on the right career path? I'm not sure math (and by extension computer programming) is really going to be your thing. There is a way of thinking and seeing the world that is required to write computer

programs that some people (no fault of their own!) just don't have. You might be happier and less frustrated doing something that is more closely related to your nature.

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.idlcoyote.com/>

Sepore ma de ni thue. ("Perhaps thou speakest truth.")

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [Moritz Fischer](#) on Tue, 03 Mar 2015 07:26:58 GMT

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

Am 02.03.2015 um 18:12 schrieb g.nacarts@gmail.com:

> Yes, I saw your last message but I am interested to understand how you end up before with the
a = reform(a, 100l, 200l*2l, /overwrite).

Ok, let's try again.

What does reform do?

It changes the arrangement of the array elements. Not its total number.

In your initial example you had an array of 200 by 200 elements (40,000 in total), and you *can* rearrange the elements of such an array to an 100 by 400 element array (also 40,000 in total), that is why I gave that example.

Helder picked the same example by chance, I might as well have suggested to reform it to 64 by 625 elements (also 40,000 in total).

Or to 100 by 16 by 25 elements.

> Then I tried to write it like this:

> a = reform(a, 120l, 216l*2l, /overwrite)

> but I got the error that I mentioned before.

Check the total element number: 120 by 216*2 = 120 by 432 = 51840

Your initial array randumu(s,216,216) has 216 by 216 = 46656 elements.

You say you want to go from 216x216 to 100x216x216.

You cannot do this with reform.

But of course you can do it differently. That would depend on what you want to look the new array like. (Copies of the original, different data

of the same size, you name it. But do name it.)

Subject: Re: REFORM: new subscripts must not change the number elements in array

Posted by [g.nacarts](#) on Tue, 03 Mar 2015 10:05:00 GMT

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

Great, now it makes sense. Thank you very much for the response and your explanation.
