Subject: Issue with long integer arrays
Posted by Daniel Otis on Tue, 17 Dec 2013 19:20:38 GMT
View Forum Message <> Reply to Message

I should know this, but am having trouble with the following:

I have several long integer(long) arrays that are restored into the workspace.

I want to define an array so that my long arrays can be used in a loop later in my program.

array1=long[25248]
array1=long[2905]
array1=long[4651]
array1=long[4988]

These are masks used to extract different regions of an image.

How can I put the arrays into one variable that can be called in a loop?

It should be simple, but I'm stumped. Thanks,

Dan

Subject: Re: Issue with long integer arrays
Posted by Helder Marchetto on Tue, 17 Dec 2013 19:42:49 GMT
View Forum Message <> Reply to Message

On Tuesday, December 17, 2013 8:20:38 PM UTC+1, do...@mail.usf.edu wrote:
> I should know this, but am having trouble with the following:
>
>
>
> I have several long integer(long) arrays that are restored into the workspace.
>
>
>
> I want to define an array so that my long arrays can be used in a loop later in my program.
>
>
>
> array1=long[25248]
>
> array1=long[2905]
>
> array1=long[4651]
>
> array1=long[4988]

>
>
>
> These are masks used to extract different regions of an image.
>
>
>
> How can I put the arrays into one variable that can be called in a loop?
>
>
>
> It should be simple, but I'm stumped. Thanks,
>
>
>
> Dan

There are more ways of doing this and depend on thing such as if you know the sizes before hand and so on. So I'll offer one and might not just be the best for your case, but I hope it helps.

```
pa = ptrarr(4,/allocate_heap)
*pa[0]=long[25248]
*pa[1]=long[2905]
*pa[2]=long[4651]
*pa[3]=long[4988]

For i=0,3 do begin
  ;your work
  print, mean(*pa[i])
endfor
```

If you know the sizes before hand, one other solution would be to make one array long as the 4 together and then subscript the array.

Cheers,
Helder

Subject: Re: Issue with long integer arrays
Posted by Daniel Otis on Tue, 17 Dec 2013 20:12:13 GMT

On Tuesday, December 17, 2013 2:42:49 PM UTC-5, Helder wrote:
> On Tuesday, December 17, 2013 8:20:38 PM UTC+1, do...@mail.usf.edu wrote:
>
>> I should know this, but am having trouble with the following:
>
>>

>
>>
>
>>
>
>> I have several long integer(long) arrays that are restored into the workspace.
>
>>
>
>>
>
>>
>
>> I want to define an array so that my long arrays can be used in a loop later in my program.
>
>>
>
>>
>
>>
>
>> array1=long[25248]
>
>>
>
>> array1=long[2905]
>
>>
>
>> array1=long[4651]
>
>>
>
>> array1=long[4988]
>
>>
>
>>
>
>>
>
>> These are masks used to extract different regions of an image.
>
>>
>
>>
>
>>

>
>> How can I put the arrays into one variable that can be called in a loop?
>
>>
>
>>
>
>>
>
>> It should be simple, but I'm stumped. Thanks,
>
>>
>
>>
>
>>
>
>> Dan
>
>
>
> There are more ways of doing this and depend on thing such as if you know the sizes before hand and so on. So I'll offer one and might not just be the best for your case, but I hope it helps.
>
>
>
> pa = ptrarr(4,/allocate_heap)
>
> *pa[0]=long[25248]
>
> *pa[1]=long[2905]
>
> *pa[2]=long[4651]
>
> *pa[3]=long[4988]
>
>
>
> For i=0,3 do begin
>
>   ;your work
>
>   print, mean(*pa[i])
>
> endfor
>
>
>

> If you know the sizes before hand, one other solution would be to make one array long as the 4 together and then subscript the array.

>

>

>

> Cheers,

>

> Helder

Thanks. Works great. Never used ptrarr.

Subject: Re: Issue with long integer arrays
Posted by Dick Jackson on Wed, 18 Dec 2013 16:17:45 GMT

On Tuesday, December 17, 2013 12:12:13 PM UTC-8, do...@mail.usf.edu wrote:

> On Tuesday, December 17, 2013 2:42:49 PM UTC-5, Helder wrote:

>

>> On Tuesday, December 17, 2013 8:20:38 PM UTC+1, do...@mail.usf.edu wrote:

>

>>> I should know this, but am having trouble with the following:

>

>>> I have several long integer(long) arrays that are restored into the workspace.

>

>>> I want to define an array so that my long arrays can be used in a loop later in my program.

>

>>> array1=long[25248]

>>> array1=long[2905]

>>> array1=long[4651]

>>> array1=long[4988]

>

>>>> These are masks used to extract different regions of an image.

>

>>> How can I put the arrays into one variable that can be called in a loop?

>

>>> It should be simple, but I'm stumped. Thanks,

>

>>> Dan

>

>> There are more ways of doing this and depend on thing such as if you know the sizes before hand and so on. So I'll offer one and might not just be the best for your case, but I hope it helps.

>

>> pa = ptrarr(4,/allocate_heap)

>> *pa[0]=long[25248]

>> *pa[1]=long[2905]

>> *pa[2]=long[4651]

>> *pa[3]=long[4988]

\>
\>> For i=0,3 do begin
\>>   ;your work
\>>   print, mean(*pa[i])
\>> endfor
\>
\>> If you know the sizes before hand, one other solution would be to make one array long as the 4 together and then subscript the array.
\>
\>> Cheers,
\>> Helder
\>
\> Thanks. Works great. Never used ptrarr.

Helder's pointer method is useful, and another method, as of IDL 8.0, is to use a list:

IDL> a=List(intarr(10), intarr(20), intarr(30)) ; Create in one statement

IDL> help,a[0]
<Expression>   INT   = Array[10]
IDL> help,a[1]
<Expression>   INT   = Array[20]
IDL> help,a[2]
<Expression>   INT   = Array[30]

IDL> a.Add, intarr(40) ; Add to list afterward

IDL> help,a[3]
<Expression>   INT   = Array[40]

You can start with an empty list (no elements) and add to it:

IDL> b = List()

Hope this helps!

Cheers,
-Dick

Dick Jackson Software Consulting
Victoria, BC, Canada --- www.d-jackson.com