Subject: Re: populating an array
Posted by David Fanning on Thu, 30 Mar 2006 19:15:04 GMT
View Forum Message <> Reply to Message

subir.vasanth@gmail.com writes:

> I was wondering if there was a more efficient way to populate an array
> created using the MAKE_ARRAY function. This is how I am populating the
> array right now -
>
> data = BYTARR(202000L)
> ; populate byte array with valid data from some input source
> array = MAKE_ARRAY(1000,101, Type = 2)
> offset = 0L
> FOR k = 0L, 100999L DO BEGIN
>   array(k) = FIX(data, offset)
>   offset = offset + 2
> ENDFOR
>
> Is there a way I can populate 'array' without using a loop to populate
> each element, and instead do a array = FIX(data)??

Say what!?

What do you think this code does, exactly? It looks to me
like it creates a 1000 by 101 array filled with integer
zeros. It would be easier to do this:

   array = IntArr(1000, 101)

But this seems so obvious, that I think I am missing
the intent of the question. Could you please elaborate
some more?

Cheers,

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

Subject: Re: populating an array
Posted by codepod on Thu, 30 Mar 2006 20:47:58 GMT
View Forum Message <> Reply to Message

Subir,

Are you trying to extract every other byte of data from your array and
then convert it to an int? If so, you could take the following
approach:

```
    data = bytarr(202000 , /nozero)
 ;; Note:  if you're filling the array later, /nozero will save a
 ;; little time (skips a bzero call).

 ;; Read your data as you mention in your message.

 ;; Here, just index into your data array using the array
 ;; stride syntax [0:*:2]   Reform creates the array shape you need.

    array = reform( fix( data[ 0:*:2 ] ), 1000, 101)

 ;; If your IDL version doesn't support strides, you can do this
 ;;  with indgen, but it will probably be slightly slower.

    array = reform( fix( data[ lindgen(101000)*2 ] ), 1000, 101)

 ;; Either of these will eleminate the costly for loop

 ;; And if your done with data at this point, you can just free it
    data = 0b
```

Cheers - CP

subir.vasanth@gmail.com wrote:
> Greetings!
>
> I was wondering if there was a more efficient way to populate an array
> created using the MAKE_ARRAY function. This is how I am populating the
> array right now -
>
> data = BYTARR(202000L)
> ; populate byte array with valid data from some input source
> array = MAKE_ARRAY(1000,101, Type = 2)
> offset = 0L
> FOR k = 0L, 100999L DO BEGIN
>   array(k) = FIX(data, offset)
>   offset = offset + 2
> ENDFOR
>
> Is there a way I can populate 'array' without using a loop to populate
> each element, and instead do a array = FIX(data)??
>

> Thanks,
> subir

Subject: Re: populating an array
Posted by subir.vasanth on Thu, 06 Apr 2006 20:43:40 GMT
View Forum Message <> Reply to Message

David,

I am trying to read in stream of bytes from a socket and then populate
an array using that data. The for loop does the job but I was trying
to get away from using the for loop and was wondering if there was a
more efficient way to populate a 2D array from a stream of bytes
(assuming you know the type of data coming in from the byte stream).

Thank you for replying!