

---

Subject: Re: Cyclic array interfaces

Posted by [David Fanning](#) on Tue, 13 Jul 2004 00:50:25 GMT

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

---

Jonathan Greenberg writes:

> A few months back I was asking about efficient ways of interacting with  
> images where I can perform spatial transforms (e.g. Applying a 5 x 5 filter  
> of some sort to the image), and I got lots of good information back, and one  
> idea in particular intrigued me: working with an array in a cyclic fashion.  
> My question is: is replacing a single "line" of an N x M array with a 1 x M  
> vector using something like:

>  
> A=[[1,2,3],[4,5,6]]  
> B=[7,8,9]  
> A[\* ,1]=B

>  
> ... An efficient method of replacing data in an array? Or does A get  
> completely rewritten and takes as much time as:

>  
> C=[[10,11,12],[13,14,15]]  
> A=C

>  
> (e.g. Does IDL actually rewrite the entire array, regardless of how many  
> elements are being changed, or does it only change the particular elements  
> and hence the first example should be about twice as fast as the second)?

>  
> If example #1 is faster than #2, then I can implement a cyclical array  
> approach, where if I want to work with 5 lines of an image at a time, the  
> "line index" the first iteration would be:

> 0,1,2,3,4

> And the second iteration (as I shift down one line):

> 5,1,2,3,4 (so I'm overwriting only one line of data at a time).

>  
> Thoughts?

Others will write with more explicit examples, probably, but here is an article that will get you started in the right direction. I'm not sure about time, but some ways of subscripting arrays can certainly take a lot of \*memory\*, which I presume means time, too.

[http://www.dfanning.com/misc\\_tips/submemory.html](http://www.dfanning.com/misc_tips/submemory.html)

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

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

---