Subject: Re: More For Loops
Posted by Liam E. Gumley on Thu, 13 Apr 2000 07:00:00 GMT
View Forum Message <> Reply to Message

majewski@cygnus.uwa.edu.au_stralia wrote:
> I'm looking to get rid of the for loops below.
> They make two arrays;
> one containing evey second column and evey second row
> the other containing evey second column and evey alternate row
>
> ;---------- ----------
> x_data = 3072
> y_data = 512
> DATA_size = [x_data/2,y_data]
> Data_sets_ev = bytarr(DATA_size[0],DATA_size[1])
> Data_sets_od = bytarr(DATA_size[0],DATA_size[1])
>
> for i = 0, (DATA_size[0]/2)-1 do begin
> for j = 0, DATA_size[1]-1 do begin
> Data_sets_ev[i,j] = my_data[(2*i),(2*j)]
> Data_sets_od[i,j] = my_data[(2*i),(2*j+1)]
> endfor
> endfor
> ;---------- ----------
>
> This is to extract a couple of NOAA14 AVHRR Sea Surface Temperature
> measurements. The different spectral bands overlap in the data file,
> hence they need to be separated by the above.

The syntax required to sample a multi-dimensional array is not
immediately obvious. For example, consider the following two dimensional
array:

```
IDL> n = 5
IDL> arr = indgen(n, n)
IDL> print, arr
      0       1       2       3       4
      5       6       7       8       9
     10      11      12      13      14
     15      16      17      18      19
     20      21      22      23      24
```

To extract every second element along each dimension, you might try
indexing the array as follows:

```
IDL> index = lindgen((n + 1) / 2) * 2
IDL> print, index
          0          2          4
```

```
IDL> print, arr[index, index]
      0      12      24
```

However this only extracts every other element along the diagonal. To extract the second element along each dimension, you must sample each dimension in turn, e.g.

```
IDL> sub = arr[index, *]
IDL> print, sub
      0       2       4
      5       7       9
     10      12      14
     15      17      19
     20      22      24
IDL> sub = sub[*, index]
IDL> print, sub
      0       2       4
     10      12      14
     20      22      24
```

With array expression indexing, the same result is obtained with a more compact syntax:

```
IDL> sub = (arr[index, *])[*, index]
IDL> print, sub
      0       2       4
     10      12      14
     20      22      24
```

Cheers,
Liam.
http://cimss.ssec.wisc.edu/~gumley
PS Say hi to MervL and NickB for me.
