Subject: array

Posted by 8sushil on Mon, 09 Mar 2015 09:27:34 GMT
View Forum Message <> Reply to Message

how can i print the location in the matrix knowing the value in that location.

Subject: Re: array

Posted by Helder Marchetto on Mon, 09 Mar 2015 09:50:25 GMT
View Forum Message <> Reply to Message

On Monday, March 9, 2015 at 10:27:39 AM UTC+1, 8sushil wrote:
> how can i print the location in the matrix knowing the value in that location.

User the where() function as described here:
http://exelisvis.com/docs/WHERE.html

The result is a 1D location. If your matrix/array is 2D and you *need* a 2D result, then you can use array_indices():
http://exelisvis.com/docs/ARRAY_INDICES.html

If the value you're looking for is myValue and the matrix/array is myArray, then you can use this:

```
position = where(myValue eq myArray, cnt)
print, 'my 1D position is: ', position
print, 'my 2D position is: ', array_indices(myArray, position)
```

Make sure you check how to use the cnt result (-1 if not found!). This should get you going. If you're looking for floats, it can be a bit more tricky.

Cheers,
Helder

Subject: Re: array

Posted by Jeremy Bailin on Mon, 09 Mar 2015 18:11:20 GMT
View Forum Message <> Reply to Message

>> how can i print the location in the matrix knowing the value in that location.
>
> User the where() function as described here:
> http://exelisvis.com/docs/WHERE.html

Also, note that if your array is sorted monotonically, VALUE_LOCATE will do it much faster.

(yes, Jeremy's back and proselytizing about VALUE_LOCATE)

-Jeremy.

Subject: Re: array
Posted by Helder Marchetto on Mon, 09 Mar 2015 18:53:58 GMT
View Forum Message <> Reply to Message

Since you're the Value_locate guy, can it be used with 2d arrays? If so, how is monotonic defined in these cases? By using 1d indexes for a 2d array?
I don't have pc access at the moment to check myself, so I thought I would ask the guru ;-)
Sorry if the question has an obvious answer.
And in case of a 2d array, when does one end up with a monotonic 2d array (however this is defined)?

Thanks,
Helder

Subject: Re: array
Posted by Jeremy Bailin on Mon, 09 Mar 2015 19:47:04 GMT
View Forum Message <> Reply to Message

> Since you're the Value_locate guy, can it be used with 2d arrays? If so, how is monotonic defined in these cases? By using 1d indexes for a 2d array?

Yes, exactly -- it needs to be monotonic with respect to 1D indices.

> And in case of a 2d array, when does one end up with a monotonic 2d array (however this is defined)?

For example:

q = indgen(N1, N2)

gives you an array that is monotonic as far as Value_Locate is concerned. Also, I will sometimes create a 2D array that is monotonic by stacking together individually-monotonic 1D arrays with offsets in the second dimension for the express purpose of using Value_Locate on it. For example, if W is an N1 x N2 array where each W[*,i] is sorted, then you can create a monotonic 2D array is follows:

```
; in order to make the array monotonic, we need to add an offset to
; each row that will ensure that the minimum value for every subsequent
; row is pushed to being greater than the maximum value of each previous
; row. The following example will technically only work for non-negative W
; and for either integers or not-too-large floats, but can be generalized.
offset = max(W)+1
```

```
Wdimen = size(W, /dimen)
; create an N1 x N2 array that adds an appropriate increment to each row
W2 = W + rebin(offset * lindgen(1, Wdimen[1]), Wdimen, /sample)
```

-Jeremy.