## Subject: array subscripting problem
Posted by m.doyle on Mon, 11 Apr 2005 13:00:14 GMT
View Forum Message <> Reply to Message

Hi Guys,

I've got a gridded dataset and I'm searching the grid trying to find those points which have a certain value. Once I've found this point, I want to check all points around it to see if they have the same value plus a little bit. This is what I've come up with:

```
 If (data(i,j,k) GE p) THEN begin

  cd = data(i,j,k)

  If(ABS(cd-data[i-1,j-1,k]) LE mg) then f = f+1
  If(ABS(cd-data[i,  j-1,k]) LE mg) then f = f+1
  If(ABS(cd-data[i+1,j-1,k]) LE mg) then f = f+1
  If(ABS(cd-data[i-1,j,  k]) LE mg) then f = f+1
  If(ABS(cd-data[i+1,j,  k]) LE mg) then f = f+1
  If(ABS(cd-data[i-1,j+1,k]) LE mg) then f = f+1
  If(ABS(cd-data[i,  j+1,k]) LE mg) then f = f+1
  If(ABS(cd-data[i+1,j+1,k]) LE mg) then f = f+1

        endif
```

However, array subscripting with negative numbers isn't allowed by IDL.

Has anyone any ideas how I might get around this?

Many thanks!

All the best
Martin

## Subject: Re: array subscripting
Posted by David Fanning on Fri, 11 Dec 2009 14:41:44 GMT
View Forum Message <> Reply to Message

Dave Higgins writes:

> I'd like to multiply the 1st dimension of a multi-dimensional array by
> a 1d vector (using * not #). I've been trying to avoid nested FOR
> loops. So far, I've come up with
>
> phi = rebin(phi, dim1,dim2,dim3,dim4,dim5)

> multi_dim_arr = multi_dim_arr * phi
>
> As you can see I'm creating a huge array from a simple "row" of data,
> so that phi and multi_dim_arr match in size. I feel like I'm missing
> something - is there an easier way to get this done?

"Easier" in what sense? To type? Make it a function. ;-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

---

## Subject: Re: array subscripting
## Posted by David Higgins on Fri, 11 Dec 2009 16:29:42 GMT

View Forum Message <> Reply to Message

On 11 Dec, 14:41, David Fanning <n...@dfanning.com> wrote:

> "Easier" in what sense? To type? Make it a function. ;-)

Thanks for your reply. Sorry I wasn't clear in my post. I was
wondering if there's a better way, in terms of memory management. I.e.
not creating a second huge array (phi after the rebin), when the
useful information is a 1d vector.

Thanks.
Dave

---

## Subject: Re: array subscripting
## Posted by David Fanning on Fri, 11 Dec 2009 16:43:48 GMT

View Forum Message <> Reply to Message

Dave Higgins writes:

> Thanks for your reply. Sorry I wasn't clear in my post. I was
> wondering if there's a better way, in terms of memory management. I.e.
> not creating a second huge array (phi after the rebin), when the

> useful information is a 1d vector.

The IDL Way often involves trade-offs between speed and memory. A judicious use of a FOR loop might be called for here, if memory usage is getting ridiculous. But I would think about transposing your array so that if you do settle on a loop solution your multiplications occur on rows of data and not columns of data. Looping on columns is sure to be slow, slow, slow.

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")