Subject: Re: using the WHERE function on a portion of an array
Posted by Jean H. on Tue, 04 Mar 2008 23:20:45 GMT
View Forum Message <> Reply to Message

becky_s wrote:
> On Mar 4, 2:15 pm, Jean H <jghas...@DELTHIS.ucalgary.ANDTHIS.ca>
> wrote:
>
>> Now you are try to apply your 2D array in a 3D one, which can not work
>> properly.
>> To access your 3D array, you must either have a 3D index, or have a 1D
>> index.
>>
>> So in your case, you want to write in C, on the 5th plane:
>> indices1D_C = indices + (n_elements(C[0,*,*]) * 4
>> And you want to read B on the 1st plane:
>> indices1D_B = indices
>>
>> and then C[indices1D_C] = B[indices1D_B]
>>
>
> Jean,
> Well, that is pretty slick! I knew there had to be some problem with
> all my 2d to 3d dimension switching I was doing.
>
> I did have to modify your solution somewhat, though. I ended up with
> (I also generalized my previous code somewhat):
>
> indices = WHERE(A[i,*,*] ge j AND A[i,*,*] lt (j+1), count)
> if count gt 0 then begin
>   indices1D_C = indices*n_elements(C[*,0,0]) + j
>   indices1D_B = indices*n_elements(A[*,0,0]) + i
>   C[indices1D_C] = B[indices1D_B]
> endif
>
> Thanks again.

Are you sure you are getting the correct indexes like that??

if you multiply the index by the number of elements, the result will be
wrong and can be out of bound!
ex: n_elements(A[*,0,0]) = 100 (10*10 array)
index = 99 (last element of the 2D array)
99 * 100 = 9 900 ==> you are in the 99th plane, even if you haven't
specified the plane!!!

on the other hand, if you know you are, let say, on the 3rd plane:
99 + (100 * (3-1)) = 299 ==> index of the last cell of the 3rd plane!

so it should really be indices1D_C = indices + n_elements(C[*,0,0]) * i

also, if you are computing this many time, you can same n_elements(C[*,0,0]) in a variable and use it each time

Jean