Subject: Re: Assignment, Form 5 (what they didn't tell you in the manuals)
Posted by steinhh on Thu, 19 Nov 1998 08:00:00 GMT
View Forum Message <> Reply to Message

In article <36534151.3A3E@bigfoot.com> David Ritscher
<davidNO.ritscherSPAM@bigfoot.com> writes:

> Today's matrix subscripting puzzler:
> I'm noting that there are some extra rules govorning assignement
> statements where the right side contains subscripting expressions in
> more than one of the dimensions of a matrix. Given a 3-D array,
> array(indgen(5), indgen(5), indgen(5)) yields a 5-element vector
> array(indgen(5), 0, indgen(5)) yields a 5 x 1 x 5 matrix
>
> I'm trying to create an index that allows me to extract vectors of
> information from a matrix that has been stored. Here is a simple
> example of the form of what I'm trying to extract. The matrix 'index'
> will put the matrix into the order I want.
>
> threeD = indgen(3,4,5)
> twoD = indgen(3,5)
> index = [[indgen(15) mod 3], [indgen(15) / 3]]
[..]

I'm not 100% sure I understand exactly what you want, so let me
rephrase the problem: Your 2d data is lindgen(m,o), and your 3d
data is lindgen(m,n,o). You rearrange your 2d data into a vector
according to the one-dimensional index value, and you want to
arrange corresponding 2d slices (*,j,*) of your 3d data into
vectors, and stack them into an array that's lonarr(m*o,n) ?

Isn't this best solved with TRANSPOSE (which has gotten some
new functionality since I learned it):

out = reform(transpose(threed,[0,2,1]),m*o,n)

?

As I said, I may have misunderstood your question, but if I
haven't, I believe this is the answer...

Stein Vidar