
Subject: Re: array transpose

Posted by [Joseph M Zawodny](#) on Tue, 22 Apr 1997 07:00:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

This is a multi-part message in MIME format.

-----17601F1B6C24

Content-Type: text/plain; charset=us-ascii

Content-Transfer-Encoding: 7bit

David Fanning wrote:

>
> Christian Soeller <csoelle@sghms.ac.uk> writes:
>
>> Gary Fu <gfu@shark.gsfc.nasa.gov> writes:
>>
>>>>
>>>> RTFM -> transpose function
>>>>
>>>> Christian
>>>
>>> The TRANSPOSE function will transpose to array(Z,Y,X), not array(Y,X,Z).
>>
>> And what about transpose(array,[1,0,2]) ?
>> As I said, it's in the manual; one obviously has to know how to read it ;).
>
> What manual are you reading, Christian? Or perhaps more to the point,
> exactly HOW are you reading it!? I don't find this is *my* manual. :-(
>
> In any case, I can't get this to work. Can you perhaps tell us what
> software you are using and give us a simple example.
>
> Thanks,
>
> David

I hope that my attachment appears below. I have written an
n-dimensional
transpose that, while not pretty or fast, does work.

Have fun,

JMZ

-----17601F1B6C24

Content-Type: text/plain; charset=us-ascii; name="reindex.pro"

Content-Transfer-Encoding: 7bit

Content-Disposition: inline; filename="reindex.pro"

```

;+
; Name: REINDEX
; Purpose: reorder the array subscripts, an n-dimensional transpose.
; Category: Array manipulation
; Calling sequence: out_arr=REINDEX(in_arr, sort)
;   or out_arr=REINDEX(/help)
;
; Example: if IN_ARR = intarr(2,3,4,5) then
; out_arr = reindex(in_arr,[2,4,3,1]) results in
; out_arr becoming an array of dimension intarr(3,5,4,2)
; or reindex( f(x,y,z),[2,3,1]) becomes f(y,z,x)
; or reindex( f(x,y,z),[1,2,3]) does nothing
;
; Inputs:
; IN_ARR n-dimensional array 1 < n < 8
; SORT vector listing order of subscripts in result
; KEYWORDS"
; /HELP  Informs the user about this procedure
;
; Output:
; OUT_ARR n-dimensional array 1 < n < 8 of same type as IN_ARR
; with subscripts reordered
;
; Optional output parameters: None
; Common blocks: None
; Side Effects: None
; Restrictions: Very large arrays may cause
; excessive page faulting
; Routines used: None
; Procedure: Straight foreward.
; Modification history:
; V-1.0 First "Public release"
; Oct 15, 1990 J.M. Zawodny, NASA LaRC
;-
function REINDEX,a,ind_order,help=help

if keyword_set(help) then begin
print,'
print,' Name: REINDEX'
print,' Purpose: reorder the array subscripts, an n-dimensional transpose.'
print,' Category: Array manipulation'
print,' Calling sequence: out_arr=REINDEX(in_arr, sort)
print,'   or out_arr=REINDEX(/help)
print,'
print,' Example: if IN_ARR = intarr(2,3,4,5) then'
print,' out_arr = reindex(in_arr,[2,4,3,1]) results in'
print,' out_arr becoming an array of dimension intarr(3,5,4,2)'

```

```

print,' or reindex( f(x,y,z),[2,3,1]) becomes f(y,z,x)'
print,' or reindex( f(x,y,z),[1,2,3]) does nothing
print,' '
print,' Inputs: '
print,' IN_ARR n-dimensional array 1 < n < 8'
print,' SORT vector listing order of subscripts in result'
print,' KEYWORDS"'
print,' /HELP  Informs the user about this procedure'
print,' '
print,' Output: '
print,' OUT_ARR n-dimensional array 1 < n < 8 of same type as IN_ARR '
print,' with subscripts reordered'
print,' '
    return,1
endif

; Inquire about input array
sa = size(a)
; Check
nind = n_elements(ind_order)

; Clone a singly dimensioned array
if(nind eq 1) then return,a

; Incompatable arrays
if(sa(0) ne nind) then begin
    print,'Number of subscripts does not match array'
    help,a,ind_order
    return,-1
endif

; Make a destination array
sb = sa([0,ind_order,nind+[1,2]])
b = make_array(size=sb)

; Compute some index arrays
i = lindgen(sa(nind+2))
c = lonarr(sa(nind+2),nind)
e = replicate(1L,nind+1)
d = 1L
for k=0,nind-1 do begin
    c(0,k) = (i/d) mod sb(k+1)
    d = d*sb(k+1)
    e(k+1) = e(k)*sa(k+1)
endfor

; A place for the sorting array
m = lonarr(sa(nind+2))

```

```
; Calculate the indicies and fill output array  
b(0)=a( c # e( ind_order( indgen(nind) )-1 ) )
```

```
; All Done  
return,b  
end
```

-----17601F1B6C24--
