Subject: Deleting from arrays
Posted by Randall Skelton on Thu, 27 Jun 2002 14:54:24 GMT
View Forum Message <> Reply to Message

Many thanks to Dick and Reimar for there excellent suggestions regarding concatenating arrays. My next question is, does anyone have code for removing data from a general (i.e. up to 8 dimensional) array? Imagine trying to remove the data marked by an asterisks '*'

In the 1D case:

```
    * * * *
[ 0 1 2 3 4 5 6 7 8 9 ]  -> result is a vector: [ 0 1 6 7 8 9 ]
```

In the 2D case:

```
  * *
  a b c d e f g                    c d e f g
*  h i j k l m n  -> result is a matrix:  q r s t u
  o p q r s t u
```

I know that Craig has code for cleverly doing the 1D case, but I'm looking to do this for at least up to 4D. I'm willing to accept that it is probably easier to do n-dimensional deletion with n separate steps. So, in the 2D case above I would first remove the dimension 0 elements (columns) followed by removing the dimension 1 elements (rows).

Something like:

```
 ; 2D case
 a = [[0,1,2,3,4], [5,6,7,8,9], [10,11,12,13,14]]
 idx = [2,3,4] ; indicies to keep in dim 0
 a = temporary( a[idx,*] )
 idx = [0,2]   ; indicies to keep in dim 1
 a = temporary( a[*,idx] )

 ; 3D case
 a = [[[0,1,2,3,4], [5,6,7,8,9], [10,11,12,13,14]],$
     [[15,16,17,18,19], [20,21,22,23,24], [25,26,27,28,29]],$
     [[30,31,32,33,34],[35,36,37,38,39],[40,41,42,43,44]]]
 idx = [2,3,4] ; indicies to keep in dim 0
 a = temporary( a[idx,*,*] )
 idx = [0,2] ; indicies to keep in dim 1
 a = temporary( a[*,idx,*] )
 idx = [1,2] ; indicies to keep in dim 2
 a = temporary( a[*,*,idx] )
```

If anyone has any ideas on how to generalize this, I'd be keen to hear

them. At the moment, I calculate the indicies of the elements I wish to keep and use different cases depending on the dimension I want to remove.

```
case dim of
  0: a = temporary( a[idx,*,*,*,*,*,*] )
  1: a = temporary( a[*,idx,*,*,*,*,*] )
  2: a = temporary( a[*,*,idx,*,*,*,*] )
  ...
endcase
```

Cheers,
Randall

PS: I am waiting for someone to post the solution as an IDL 'one-liner' ;)
