
Subject: Re: arrays with different sizes

Posted by [Michael Galloy](#) on Fri, 20 May 2011 22:03:57 GMT

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

On 5/20/11 3:31 PM, Nolan Smith wrote:

> Hello,
>
> I am new to IDL and I have the following problem.
> I have 3 different array with different sizes
> arr1=[2,3,4,5,6,10]
> arr2=[1,2,5,6]
> arr3=[4,5,7,8,9,9,12,15]
> and I am trying to create an array that will have all the different
> permutations of the above array elements.
> In the end I want to have an array that will look like this:
> 2 1 4
> 2 1 5
> 2 1 7
>
>
> 10 6 12
> 10 6 15
>
> I can not figure out a way to do this, any ideas?
> My arrays are much larger than those in the example so a quick
> solution will be appreciated too!
>
> Thank you!

How about this?

```
function mg_combo3, arr1, arr2, arr3
  compile_opt strictarr

  n1 = n_elements(arr1)
  n2 = n_elements(arr2)
  n3 = n_elements(arr3)

  ind1 = reform(rebin(reform(lindgen(n1), n1, 1, 1), $
    n1, n2, n3), $
    n1 * n2 * n3)
  ind2 = reform(rebin(reform(lindgen(n2), 1, n2, 1), $
    n1, n2, n3), $
    n1 * n2 * n3)
  ind3 = reform(rebin(reform(lindgen(n3), 1, 1, n3), $
    n1, n2, n3), $
    n1 * n2 * n3)
```

```
result = lonarr(3, n1 * n2 * n3)
result[0, *] = arr1[ind1]
result[1, *] = arr2[ind2]
result[2, *] = arr3[ind3]
```

```
return, result
end
```

```
[121]> print, mg_combo3(indgen(2), indgen(3), indgen(4))
```

0	0	0
1	0	0
0	1	0
1	1	0
0	2	0
1	2	0
0	0	1
1	0	1
0	1	1
1	1	1
0	2	1
1	2	1
0	0	2
1	0	2
0	1	2
1	1	2
0	2	2
1	2	2
0	0	3
1	0	3
0	1	3
1	1	3
0	2	3
1	2	3

Mike

--

Michael Galloy

www.michaelgalloy.com

Modern IDL, A Guide to Learning IDL: <http://modernidl.idldev.com>

Research Mathematician

Tech-X Corporation
