
Subject: Re: Efficient comparison of arrays
Posted by [wonko](#) on Mon, 11 Aug 1997 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

afl@cdc.noaa.gov (Andy Loughe) wrote:

- > Given vectors of the type...
- >
- > a = [1,2,3,4,5]
- > b = [3,4,5,6,7]
- >
- > What is the most efficient way to determine which values that occur in
- > a also occur in b (i.e., the values [3,4,5] occur in both a and b).
- >
- > Presumably this needs to be done without loops (to be efficient), but
- > an obvious solution escapes me.

What type of data would this be? If you are dealing with integers, what about this:

```
max_value = max( [ a, b ], min_value=min_value )
mask = indgen( max_value-min_value+1 )
mask(a-min_value) = 1
mask(b-min_value) = mask(b-min_value) + 1
values = where( mask eq 2, count ) + min_value
```

Voila, no loops, I would give this a try. Bad luck if you have floating point data.

Someone already came up with this question five months ago, but I don't remember the responses. Maybe you can find the thread via dejanews.

Alex

--

Alex Schuster Wonko@weird.cologne.de PGP Key available
alex@pet.mpin-koeln.mpg.de
