Subject: Re: Removing equal elements from an array
Posted by JD Smith on Thu, 17 Aug 2006 17:59:05 GMT
View Forum Message <> Reply to Message

On Thu, 17 Aug 2006 10:07:59 -0600, R.G. Stockwell wrote:

>
> "Julio" <julio@cpa.unicamp.br> wrote in message
> news:1155653196.084429.65000@74g2000cwt.googlegroups.com...
>> [quoted text muted]
>
> If I follow you correctly, you want to find unique pairs of numbers right?
> How about combining the pairs into one number, and running uniq() on that?
>
> For instance:
>
> A = fltarr(2,4)
> A[0,*]=[20.4, 40.3, 50.2, 50.2]
> A[1,*]=[30.2, 60.2, 32.4, 32.4]
> combo = A[0,*]*1000 + A[1,*]
> indices = UNIQ(combo, SORT(combo))

This method is so deceptively simple that it really seems like it should work, and sometimes it does, but it doesn't, in general, for floating point numbers, even if they fall in the proper range. I gave a similar example before, but it's worth repeating:

```
[.2,10]  => combo=210
[.1,110] => combo=210
[.15,60] => combo=210
```

or what about:

```
[5.5,5.5]  => combo= 5505.5
[5.2,305.] => combo= 5505.5
```

The only sure fire way to map a given pair of numbers over a finite range to a single unique combination, is to scale them to integers, then offset one set entirely from the other (64bit integers are nice for this). For small sets of numbers you might not run into this form of collision, but for many numbers, it's quite likely you would.

JD