
Subject: Re: Ordered index array

Posted by [David Fanning](#) on Thu, 08 Sep 2005 13:02:36 GMT

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

Paolo Grigis writes:

```
> But this will fail if 'a' has more elements than the number of
> its different values, for instance a=[3,3,1,2,3,2,2,1,2,3].
>
> One could try this:
>
> PRO test
>
> a=[3,3,1,2,3,2,2,1,2,3]
>
> b=a
> c=intarr(n_elements(a))
>
> h=histogram(a,min=1,reverse_ind=ri)
>
> done=0
> rank=1
> WHILE NOT done DO BEGIN
>   actual_value=b[0]
>   ind_actual_value=ri[ri[actual_value-1]:ri[actual_value]-1]
>   c[ind_actual_value]=rank
>   rank=rank+1
>   indremove=where(b NE actual_value,count)
>   IF count GT 0 THEN b=b[indremove] ELSE done=1
> ENDWHILE
>
> print,a
> print,c
>
> END
>
> but it will get inefficient as the numbers of different values
> in 'a' grows, as the code in the loop get called more and more
> times...
```

Humm. Maybe I'm missing something (I haven't had my coffee yet), but couldn't this be solved with the original code I wrote by changing these two lines:

```
; Create index array and output array.
b = Indgen(N_Elements(a))+ 1
c = Intarr(N_Elements(a))
```

To these:

```
; Create index array and output array.  
b = Indgen(N_Elements(a) > N_Elements(h)) + 1  
c = Intarr(N_Elements(a) > N_Elements(h))
```

Then,

```
IDL> a=[3,3,1,2,3,2,2,1,2,3]  
IDL> Print, a  
1 1 3 4 1 4 4 3 4 1
```

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
