Subject: Re: What? You can't histogram a string array?
Posted by JD Smith on Tue, 28 Nov 2006 18:08:24 GMT
View Forum Message <> Reply to Message

On Tue, 28 Nov 2006 09:16:12 -0800, Braedley wrote:

> JD, a small nitpick: ind_int_sort will occasionally take the index from
> [a, b], and not from just a. This can quickly lead to out of bounds
> conditions if the user doesn't want to index [a, b], but just wants to
> index a. In my case, a is a column from a 2D string array, where b is
> just a 1D string array. I think a where statement is all that is
> needed to fix this (I know, it'll slow it down for large sets).

This is not good, and much worse than a minor nitpick. The IND_INT_SORT algorithm relies on SORT doing the right thing. That is, for two identical elements in the concatenated vector [a,b], SORT should place the first one first, i.e. the matching elements from 'a' will show up before those from 'b'. That's the only reason it works. There was always the concern that IDL's SORT would change and this would no longer be the case (the element from b would come first), in which case the algorithm would be broken.

Can you provide an example where this isn't happening? I just tried it on a simulated set of 100,000 random 6 character strings, and it didn't show this behavior: all ~30 matching elements were selected from a. I then ran this test 100 times, and in all cases it behaved as expected. Perhaps it depends on the machine/OS? I'm actually not sure if SORT calls a library sort function (which might make the algorithm non-portable), or uses its own. You can try this test yourself, like this:

```
for i=1,100 do begin
  a=string(byte(randomu(sd,6,100000)*26)+65b)
  b=string(byte(randomu(sd,6,100000)*26)+65b)
  s=ind_int_sort(a,b)
  print,strtrim(n_elements(s),2),' matches found'
  m=max(s)
  if m ge 100000 then begin
    print,'Out of bounds: ',m
    break
  endif
endfor
```

Let me know if it runs through without error for you. For anyone else who wants to test this, it would be appreciated. Here I run:

```
IDL> help,!VERSION,/st
** Structure !VERSION, 8 tags, length=76, data length=76:
```

```
ARCH            STRING    'x86'
OS              STRING    'linux'
OS_FAMILY       STRING    'unix'
OS_NAME         STRING    'linux'
RELEASE         STRING    '6.3'
BUILD_DATE      STRING    'Mar 23 2006'
MEMORY_BITS     INT             32
FILE_OFFSET_BITS
                INT             64
```

BTW, if you only want the *values*, not the positions, where match occurred, replace:

return,srt[wh]

with

return,s[wh]

and this will "solve" the problem for you (with this change, it's equivalent to the CONTAIN function I posted long long ago). This is insensitive to the ordering of a or b SORT performs.

Also note that IND_INT_SORT only returns *one* match for repeated elements, which may or may not be what you want.

JD
