Subject: Re: Finding strings values common to two (large!) arrays
Posted by rjp23 on Wed, 07 Oct 2015 19:57:19 GMT
View Forum Message <> Reply to Message

CMSET_OP looks to be working but I'm not 100% sure due to this comment in the header:

```
;   INDEX - if set, then return a list of indices instead of the array
;           values themselves.  The "slower" set operations are always
;           performed in this case.
;
;           The indices refer to the *combined* array [A,B].  To
;           clarify, in the following call: I = CMSET_OP(..., /INDEX);
;           returned values from 0 to NA-1 refer to A[I], and values
;           from NA to NA+NB-1 refer to B[I-NA].
```

When using the code like this, it is returning an array of indices that only seem to relate to the first array.

e.g. (massively simplied) A has 10 elements, B has 20 and the returned indices are an array of 7 values such as [0,1,2,5,7,8,9]

Would I not also expect indices for the elements in the second array (10-29) to also be returned going by the statement in the header?

On Wednesday, October 7, 2015 at 4:21:31 PM UTC+1, rj...@le.ac.uk wrote:
> The IDs are of the form: 200904230023043018 0019
>
> I think that's too long to convert into a number (at least when I try to turn it into a long it ends up very different!)
>
> CMSET_OP looks like it's what I need. Thanks both :-)
>
>
> On Wednesday, October 7, 2015 at 4:09:29 PM UTC+1, wlandsman wrote:
>> Two points to consider:
>>
>> I second Helder's suggestions but have two additional points to consider:
>>
>> 1.  Do your array A have duplicate values?  And if so, do you want to find the indices of all the values, even if they are repeated?  Then I would suggest using
>>
>> http://idlastro.gsfc.nasa.gov/ftp/pro/misc/match2.pro
>>
>> which will return every matching index even of duplicate values.
>>

>> 2. You say the arrays are "numerical IDs in string format". Are you able to convert these strings into numerical values? If so, the matching algorithms work faster for numerical arrays (especially integers) than for strings. I do suspect the speed difference is not important unless you have to do the matching many times.
>>
>> --Wayne
>>
>> On Wednesday, October 7, 2015 at 10:45:55 AM UTC-4, Helder wrote:
>>> On Wednesday, October 7, 2015 at 4:13:59 PM UTC+2, rj...@le.ac.uk wrote:
>>>> I have arrays of numerical IDs in string format.
>>>>
>>>> I want to find all of the indices in Array A that contain a value that is present anywhere in Array B.
>>>>
>>>> The arrays are both quite large (>1 million values) so a loop is out of the question and them being strings complicates it as well.
>>>>
>>>> Any IDL Way tips?
>>>
>>> Interesting... I guess that a set operation will do or in other words, you want to find (A) AND (B)
>>> Did you look at David's page:
>>> https://www.idlcoyote.com/tips/set_operations.html
>>>
>>> There are some good tips, among which Craig's CMSET_OP which works also on strings (but does not return indices...).
>>>
>>> I hope it helps.
>>>
>>> Cheers,
>>> Helder

---