Subject: Re: Array comparison
Posted by mchinand on Wed, 02 Oct 2002 00:07:42 GMT
View Forum Message <> Reply to Message

In article <and6i0$8l4$1@newsreader.wustl.edu>,
Sean Raffuse <sean@me.wustl.edu> wrote:
> Hello,
>
> I would like to compare two arrays of different size. What I want to know
> is if the two arrays share ANY of the same values, regardless of index.
>
> e.g.
>
> Array1 = [1,2,3,4,5]
> Array2 = [5,6,7]
>
> Comparing these two arrays should return true because they both have the
> value 5.
>
> I know I could do this in a loop, but I need the speed and I'm sure IDL can
> do this in one line. I'm just not sure how.
>
> Thanks,
>
> Sean Raffuse
>
>

How about:

if n_elements(uniq(([a,b])(sort([a,b])))) ne n_elements([a,b]) then print,
'arrays have matching values'

--Mike

--
Michael Chinander
m-chinander@uchicago.edu
Department of Radiology
University of Chicago

Subject: Re: Array comparison
Posted by Craig Markwardt on Wed, 02 Oct 2002 02:59:56 GMT
View Forum Message <> Reply to Message

"Sean Raffuse" <sean@me.wustl.edu> writes:

> I would like to compare two arrays of different size. What I want to know
> is if the two arrays share ANY of the same values, regardless of index.
>
> e.g.
>
> Array1 = [1,2,3,4,5]
> Array2 = [5,6,7]
>
> Comparing these two arrays should return true because they both have the
> value 5.

Mike has a nice one-off comparison you can use. For your kinds of
set-comparison tests, I use CMSET_OP with an intersection test.

```
intersection = cmset_op(array1, 'AND', array2, count=ct)
if ct GT 0 then print, 'Arrays match'
```

The variable INTERSECTION contains the intersection between the two
sets. The internals of CMSET_OP are ultimately the same as Mike's
approach, but with a bit more flexibility.

Craig

http://cow.physics.wisc.edu/~craigm/idl/idl.html

--

---------- ----------
Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
---------- ----------

Subject: Re: Array comparison
Posted by R.Bauer on Wed, 02 Oct 2002 08:11:33 GMT
View Forum Message <> Reply to Message

Sean Raffuse wrote:
> Hello,
>
> I would like to compare two arrays of different size. What I want to know
> is if the two arrays share ANY of the same values, regardless of index.
>
> e.g.
>
> Array1 = [1,2,3,4,5]
> Array2 = [5,6,7]

>
> Comparing these two arrays should return true because they both have the
> value 5.
>
> I know I could do this in a loop, but I need the speed and I'm sure IDL can
> do this in one line. I'm just not sure how.
>
> Thanks,
>
> Sean Raffuse
>
>

I am using which_indices

http://www.fz-juelich.de/icg/icg-i/idl_icglib/idl_source/idl
_html/dbase/download/which_indices.tar.gz
or as idl5.5 binary
http://www.fz-juelich.de/icg/icg-i/idl_icglib/idl_source/idl _html/dbase/download/which_indices.sav

```
ix=which_indices(Array1,Array2)
print,array1[ix]
  5
```

regards

Reimar

--
Reimar Bauer

Institut fuer Stratosphaerische Chemie (ICG-I)
Forschungszentrum Juelich
email: R.Bauer@fz-juelich.de
---------- -------
a IDL library at ForschungsZentrum Juelich
http://www.fz-juelich.de/icg/icg-i/idl_icglib/idl_lib_intro. html
========== =======

Subject: Re: Array comparison
Posted by Craig Markwardt on Wed, 02 Oct 2002 13:19:08 GMT
View Forum Message <> Reply to Message

Reimar Bauer <R.Bauer@fz-juelich.de> writes:

> Sean Raffuse wrote:
>> Hello,
>>
>> I would like to compare two arrays of different size. What I want to know
>> is if the two arrays share ANY of the same values, regardless of index.
>>
>> e.g.
>>
>> Array1 = [1,2,3,4,5]
>> Array2 = [5,6,7]
>>
>> Comparing these two arrays should return true because they both have the
>> value 5.
>>
>> I know I could do this in a loop, but I need the speed and I'm sure IDL can
>> do this in one line. I'm just not sure how.
[...]
>
> I am using which_indices

Greetings Reimar--

I believe that the original poster wanted was the values, and not the indices.

Here is what WHICH_INDICES uses:

```
>   FOR i=0L,n_clients DO BEGIN
>     a=WHERE(master EQ client[i],count_a)
>     IF count_a GT 0 THEN build_vector,result,a
>   ENDFOR
```

I am not sure that the FOR loop with an interior WHERE() will scale to very large arrays, or very fast calculations, as the original poster appeared to be requesting.

This actually ties into JD's point from the other day. A loop of WHERE()'s can be quite slow, even slower than a loop of basic operations. Its speed here is roughly proportional to N*M, where N and M are the two input vector sizes. The SORT/UNIQ method is proportional to (N+M)*ALOG(N+M), which can be a big savings for large arrays.

Craig

--
---------- ----------

Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
---

Subject: Re: Array comparison
Posted by Sean Raffuse on Wed, 02 Oct 2002 14:34:28 GMT
View Forum Message <> Reply to Message

>
> I believe that the original poster wanted was the values, and not the
> indices.
>

That is indeed true. Actually, I'm not interested in anything but whether or not an intersection exists. Neither the values nor the indices are important.

I am not doing this on very large arrays. However, I may be doing it many, many times.

Thanks,

Sean

Subject: Re: Array comparison
Posted by David Fanning on Wed, 02 Oct 2002 14:53:47 GMT
View Forum Message <> Reply to Message

Sean Raffuse (sean@me.wustl.edu) writes:

> That is indeed true. Actually, I'm not interested in anything but whether
> or not an intersection exists. Neither the values nor the indices are
> important.
>
> I am not doing this on very large arrays. However, I may be doing it many,
> many times.

I've always found the set intersection, union, and difference routines on my web page useful for these kinds of operations. I've modified them some to do better error checking, etc., but this only takes a couple of seconds.

http://www.dfanning.com/tips/set_operations.html

Cheers,

David
--
David W. Fanning, Ph.D.
Fanning Software Consulting, Inc.
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Array comparison
Posted by R.Bauer on Wed, 02 Oct 2002 15:09:27 GMT
View Forum Message <> Reply to Message

Craig Markwardt wrote:
> Reimar Bauer <R.Bauer@fz-juelich.de> writes:
>
>
>> Sean Raffuse wrote:
>>
>>> Hello,
>>>
>>> I would like to compare two arrays of different size. What I want to know
>>> is if the two arrays share ANY of the same values, regardless of index.
>>>
>>> e.g.
>>>
>>> Array1 = [1,2,3,4,5]
>>> Array2 = [5,6,7]
>>>
>>> Comparing these two arrays should return true because they both have the
>>> value 5.
>>>
>>> I know I could do this in a loop, but I need the speed and I'm sure IDL can
>>> do this in one line. I'm just not sure how.
>>
> [...]
>
>> I am using which_indices
>
>
>
> Greetings Reimar--
>
> I believe that the original poster wanted was the values, and not the
> indices.

Ok, it looks only similiar to another FAQ Question so I did not read the whole mail. My mistake. Sorry

best wishes

Reimar

--
Reimar Bauer

Institut fuer Stratosphaerische Chemie (ICG-I)
Forschungszentrum Juelich
email: R.Bauer@fz-juelich.de
---------- -------
a IDL library at ForschungsZentrum Juelich
http://www.fz-juelich.de/icg/icg-i/idl_icglib/idl_lib_intro. html
========== =======

Subject: Re: Array comparison
Posted by Craig Markwardt on Wed, 02 Oct 2002 15:12:01 GMT
View Forum Message <> Reply to Message

"Sean Raffuse" <sean@me.wustl.edu> writes:

>>
>> I believe that the original poster wanted was the values, and not the
>> indices.
>>
>
> That is indeed true. Actually, I'm not interested in anything but whether
> or not an intersection exists. Neither the values nor the indices are
> important.
>
> I am not doing this on very large arrays. However, I may be doing it many,
> many times.

Another way to do set comparisons is with HISTOGRAM. This method is used within the set_operations.pro file that David Fanning just mentioned. It is also used by CMSET_OP, if the dynamic range of values is small. If the dynamic range is large, then there is a penalty because a huge histogram is created to store a few values. CMSET_OP chooses dynamically whether to use the HISTOGRAM or SORT/UNIQ.

Craig

--
---------- ----------
Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
---------- ----------

Subject: Re: Array comparison
Posted by Jeff Guerber on Wed, 02 Oct 2002 19:04:57 GMT
View Forum Message <> Reply to Message

On Wed, 2 Oct 2002, Mike Chinander wrote:

> How about:
>
> if n_elements(uniq(([a,b])(sort([a,b])))) ne n_elements([a,b]) then print,
> 'arrays have matching values'

Uhh, doesn't that claim a match if one of the arrays contains two (or more) identical elements, even if the arrays have no elements in common?

```
IDL> a=[1,2,3,4,5]
IDL> b=[6,6,7]
IDL> if n_elements(uniq(([a,b])(sort([a,b])))) ne n_elements([a,b]) then
print,'match'
match
IDL>
```

Jeff Guerber

Subject: Re: Array comparison
Posted by mchinand on Wed, 02 Oct 2002 20:44:52 GMT
View Forum Message <> Reply to Message

In article <Pine.GHP.4.32.0210021447020.18221-100000@icesat2.gsfc.nasa.gov>,
Jeff Guerber <jeff.guerber@gsfc.nasa.gov> wrote:
> On Wed, 2 Oct 2002, Mike Chinander wrote:
>
>> How about:
>>
>> if n_elements(uniq(([a,b])(sort([a,b])))) ne n_elements([a,b]) then print,
>> 'arrays have matching values'
>
> Uhh, doesn't that claim a match if one of the arrays contains two (or
> more) identical elements, even if the arrays have no elements in common?

>
> IDL> a=[1,2,3,4,5]
> IDL> b=[6,6,7]
> IDL> if n_elements(uniq(([a,b])(sort([a,b])))) ne n_elements([a,b]) then
> print,'match'
> match
> IDL>
>
> Jeff Guerber
>
Yeah, I realized this after reading the code from David Fanning's site
that he linked to in this thread. Applying UNIQ to the two arrays prior
to concatenating them should get around this.

--Mike

--
Michael Chinander
m-chinander@uchicago.edu
Department of Radiology
University of Chicago
