Subject: Undocumented array indexing feature?
Posted by Liam Gumley on Thu, 21 Jan 1999 08:00:00 GMT
View Forum Message <> Reply to Message

Here's a way to index an array I hadn't seen before:

```
a = indgen(10,10)
x = [3,5,8,9]
y = [2,7]
print, (a[x,*])[*,y]
      23      25      28      29
      73      75      78      79
```

Does anyone know where this feature (i.e. enclosing an array with parentheses and appending an index) is documented? I couldn't find it in my printed IDL 5.0 documentation.

Cheers,
Liam.

Liam E. Gumley
Space Science and Engineering Center, UW-Madison
1225 W. Dayton St., Madison WI 53706, USA
Phone (608) 265-5358, Fax (608) 262-5974
http://cimss.ssec.wisc.edu/~gumley

Subject: Re: Undocumented array indexing feature?
Posted by Martin Schultz on Tue, 26 Jan 1999 08:00:00 GMT
View Forum Message <> Reply to Message

R.Bauer wrote:
>> Bill B. wrote:
> [...]
>> Oh, I knew that one, Alex. Anyone know which method offers faster execution -
>>
>> a = 0 & b = 0 * OR * a = (b = 0)
>>
>> this is important when applying the same principle to large arrays. At least
>> to those of us with slower PCs :(
>
> Initializing of big arrays should be done by a=make_array(200,100000,/nozero)
> This is the fastest method
>
> R.Bauer

but then you have them uninitialized! A fair comparison is

```
  a = make_array(dim1,dim2,type)

Here is some timing info (IDL 5.1 on SGI Origin 2000):
[only one pass each, but these numbers don't change much]

A1 = FLTARR(1000,1000) & A2 = FLTARR(1000,1000)
     0.031656981 +      0.032227993 =      0.063884974 seconds

B2 = ( B1 = FLTARR(1000,1000) )
     0.085137010 seconds

C1 = MAKE_ARRAY(1000,1000,/FLOAT) & C2 = ...
     0.037186027 +      0.040596962 =      0.077782989 seconds

D1 = MAKE_ARRAY(1000,1000,/FLOAT,/NOZERO) & C2 = ...
   0.00017607212 +    0.00011897087 =    0.00029504299 seconds


Regards,
Martin.

--
 ------------------------------------------------------------ -------
Dr. Martin Schultz
Department for Engineering&Applied Sciences, Harvard University
109 Pierce Hall, 29 Oxford St., Cambridge, MA-02138, USA

phone: (617)-496-8318
fax  : (617)-495-4551

e-mail: mgs@io.harvard.edu
Internet-homepage: http://www-as.harvard.edu/people/staff/mgs/
 ------------------------------------------------------------ -------
```

Subject: Re: Undocumented array indexing feature?
Posted by R.Bauer on Tue, 26 Jan 1999 08:00:00 GMT

View Forum Message <> Reply to Message

wbiagiot@suffolk.lib.ny.us wrote:

```
>  In article <36AC8341.EC8@rosa.mpin-koeln.mpg.de>,
>    Alex Schuster <alex@rosa.mpin-koeln.mpg.de> wrote:
>>  wrb1000@my-dejanews.com wrote:
>>
>>>  This is veering offtopic, but I'd just like to add my two cents.
>>>  IDL, like C, has many constructs that may add placed on a line to
>>>  condense the actual length of the program.  However, just like C, the
```

>>> readablity and understanding factors tend to drop. I intend to use the
>>> a=(b=(c=1)) example. I always wondered how to initialize multiple variables
>>> on the same IDL line.
>>
>> Now that's an easy one:
>>
>> IDL> a=1 & b=2 & c=3 & d=4 &
>>
>> Voila, one line :-)
>>
>> Alex
>
> Oh, I knew that one, Alex. Anyone know which method offers faster execution -
>
> a = 0 & b = 0 * OR * a = (b = 0)
>
> this is important when applying the same principle to large arrays. At least
> to those of us with slower PCs :(

Initializing of big arrays should be done by a=make_array(200,100000,/nozero)
This is the fastest method

R.Bauer
