
Subject: Re: Array subscripting oddity
Posted by [offenbrg](#) on Tue, 17 Aug 1993 16:40:45 GMT
[View Forum Message](#) <> [Reply to Message](#)

glenn@atm.ch.cam.ac.uk (Glenn Carver) writes:

```
> WAVE> a=indgen(3)
> WAVE> print,a
>   0   1   2
> WAVE> m=[45,99]
> WAVE> print,m
>   45   99
> WAVE> a(m)=255
> WAVE> print,a
>   0   1  255
```

> Now, I would have thought that because both elements of 'm' are more
> than the size of 'a', no element of 'a' would have been assigned?
> The values '45' and '99' are completely arbitrary. You can set them to
> anything you like.

I'm speaking from IDL experience, not PV-WAVE, but here goes:

IDL isn't like C. In C, if A is allocated as a 3-element vector, you can still read from or write to A[45] (albeit with unpredictable results :).

In IDL an array isn't exactly a pointer like it is in C. If you try to access an element outside of the range of the array, you get an error message. For example, if you said "a(45) = 255", you'd get an "array subscript out of range" (or some such) message.

When the subscript (m in the example) is a vector, IDL treats it differently. Essentially, it treats out-of-bounds elements *when the subscript is a vector* as if they were the nearest element (either element 0 or the last element).

Example:

```
IDL> a = indgen(3) & print,a
0 1 2
IDL> print, a(-1) & print, a(5)
% Attempt to subscript A with <INT ( -1)> is out of range.
% Execution halted at $MAIN$ .
% Attempt to subscript A with <INT ( 5)> is out of range.
% Execution halted at $MAIN$ .
```

```
IDL> print, a([-1,5]) ;The subscript is now a vector
0 2
```

I'm not quite sure why it was done this way--I think it is because one can treat A as a function, and this way A doesn't return null values, and the result has the same number of elements as the input.

e.g.

IDL> F = A(m) ;F has the same dimensions as m, even if individual elements
;of m are out of range for A.

Joel

--

Joel D. Offenberg |
Hughes STX / UIT Science Team | I get paid to stare into space.
offenbrg@fondue.gsfc.nasa.gov |
Don't hold anyone responsible for what I say. I'm on my coffee break.
